28 Feb 2018

The right & wrong ways

On the 23rd Feb 2018 I commented on a LinkedIn article called “Data pros waste half of their work time chasing costly data” in which I summarised the sequence in which I declare the steps that need to be undertaken if this waste of time was to be avoided. These steps were discovered after some 20 years of research and development into both the business and technology domains. The approaches researched were:
1) Set theory
2) IBM's management by objectives theories
3) Peter Drucker's Strategic planning theories
4) Ed Yourdon's Structured analysis and structured design methodologies
5) Charlie Bachman's CODASYL theories
6) IBM's D/L 1 (IBM's hierarchical database language) theories
7) Edger Codd's normalisation methodology
8) MA Jackson's Structured Programming approach
9) James Martin & Clive Finklestein's information engineering methodology

Each had their strengths, but all had glaring weaknesses. None of them:
1) Produced explicit enough deliverables
2) Started the process with the right activity
3) Produced explicit business nor technical models
4) Seamlessly integrated the deliverables. They were at best implicit and at worst unintegrated
5) Were cost effective. That is none of them could produce a cost-benefit analysis

I would now like to introduce the major components of both the business and technology domains and concentrate on where they come from, what they deliver and how inefficient and ineffective they are in comparison to The Ripose Technique. This also provides the capability to compare any approach with any other.

I came to this conclusion as every one of these so-called ‘best practice’ approaches has had to have had their origins in and built on one or more of the aforementioned 9 approaches, yet they continue to emulate the 5 weaknesses.

My first eight examples are those of: The Open Group Architecture Framework; The Zachman Framework; The Object Management Group; An implementation of: a) Design Thinking; b) A Canvas model; c) Innovation management; Data modelling; and PEAF.

I may add additional approaches should anyone request me to.

Charles Meyer Richter
Principal information architect and diagnostician
Ripose Pty Limited
charles.richter@ripose.com

Revised 18 Mar 2018 E&OE

Ripose Technique

According to my empirical research, the hierarchical subordination of the domains (mentioned on the previous page) which will prevent the waste of valuable resources (people, time and money) is as follows:

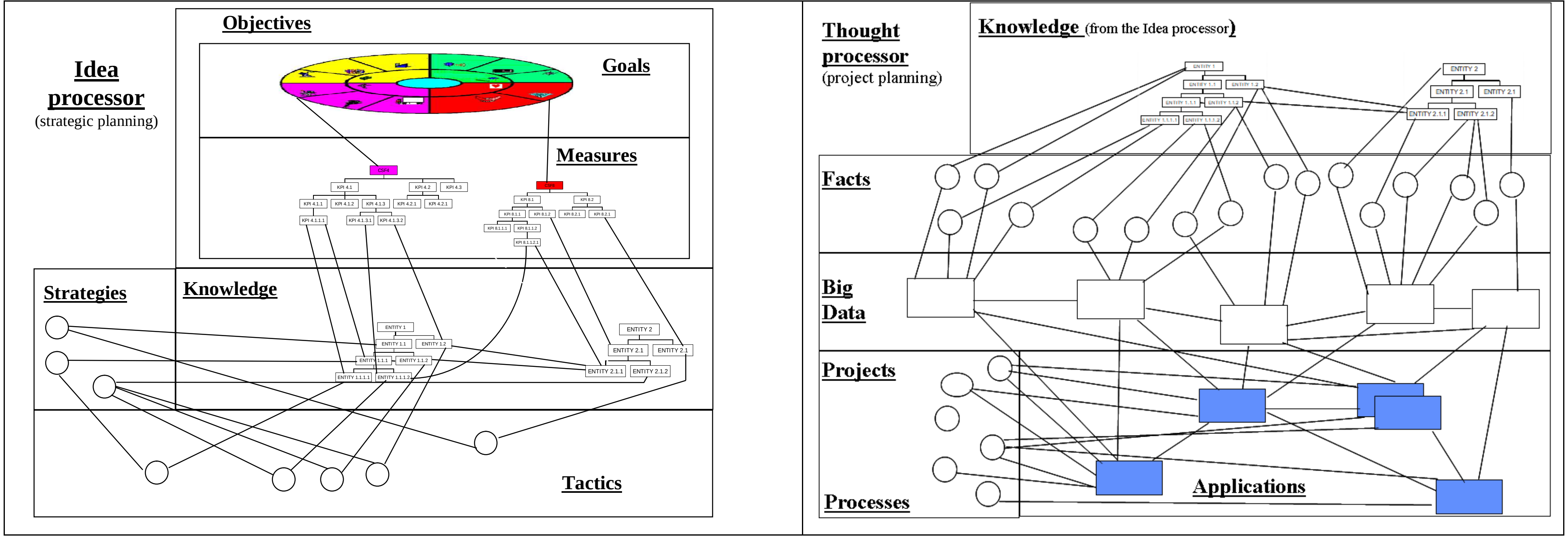
Tabular view

Domain	Step	Focus	Deliverables	State	Links
Business	1	Information	All - *Proof of Concept, Proof of Logic & Proof of Physical*	Implicit	Dreams
	1.1	Concepts	Objectives, Knowledge, & Actions - *Proof of Concept*		Information
	1.1.1	Objectives	Goals, Measures		Concepts
	1.1.1.1	Goals	Purpose, Benefits, Values		Objectives
	1.1.1.1.1	Purpose statement	Purpose	Explicit	Goals
	1.1.1.1.2	Benefits	4 benefits, 4 hardships		Purpose
	1.1.1.1.3	Values	11 values, 11 de-values		Benefits
	1.1.1.1.4	SWOT	SWOT analysis		Values
	1.1.1.2	Measures	Cost benefit analysis		Benefits
	1.1.1.2.1	Key performance indicators (KPIs)	KPIs prioritised by result from the SWOT analysis		Values
	1.1.1.2.2	Performance indicators (PIs)	Subordinate PIs – income and expense streams		KPIs
	1.1.2	Knowledge - industry specific	Entities - 23 fundamental classes & 350+ networked		PIs
	1.1.3	Actions	Strategies/Systems and tactics/sub-systems	Implicit	Knowledge
	1.1.3.1	Systems	5 generic strategies	Explicit	
	1.1.3.2	Sub-systems Industry dependent	Variable sub-systems – could be between 10 and 100		Systems
Technology	1.2	Logic	Facts, Projects & Applications – *Proof of Logic*	Implicit	Sub-systems
	1.2.1	Facts	Data & Databases		
	1.2.1.1	Data	Attributes	Explicit	Knowledge Sub-systems
	1.2.1.2	Databases	Database schemas		Data
	1.2.2	Projects	Prioritised project plans (Subject areas)		Databases
	1.2.3	Applications	Program logic		Projects
	1.3	Physical	Database definitions & Programs – *Proof of Physical*	Implicit	Logical
	1.3.1	Database definitions	Database management specific designs	Explicit	Databases
	1.3.2	Programs	Language specific computer generated code		Applications

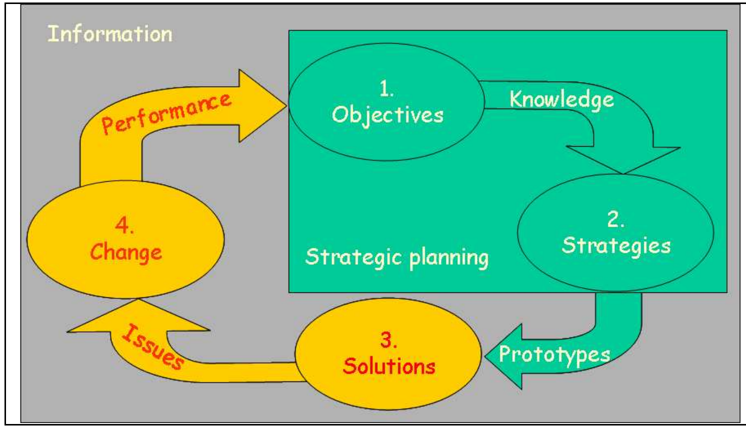
This provides me with a benchmark standard against which I can now compare Ripose to any other approach and any other approach to any other. In addition it enables a Ripose grade 0 (or any other Ripose grade) to work with any other approach and deliver the Ripose deliverables. But why bother?

Once you have looked at the comparison between Ripose and TOGAF and between Ripose and Zachman you can ten see the comparison between TOGAF and Zachman

Diagrammatic view



Meta view



Legend: Green Ripose Technique supported by Caspar
Yellow Platform dependent

Get this wrong & enjoy the wasted time trying to evade the 2 traps, namely ‘analysis by paralysis’ and ‘a death by 1,000 cuts’.

The next few pages will demonstrate the problems a number of comparative approaches face due to the implicit nature of their deliverables and the sequence in which they prioritise their steps.

Back

Comparisons with other approaches

1. TOGAF

Tabular view

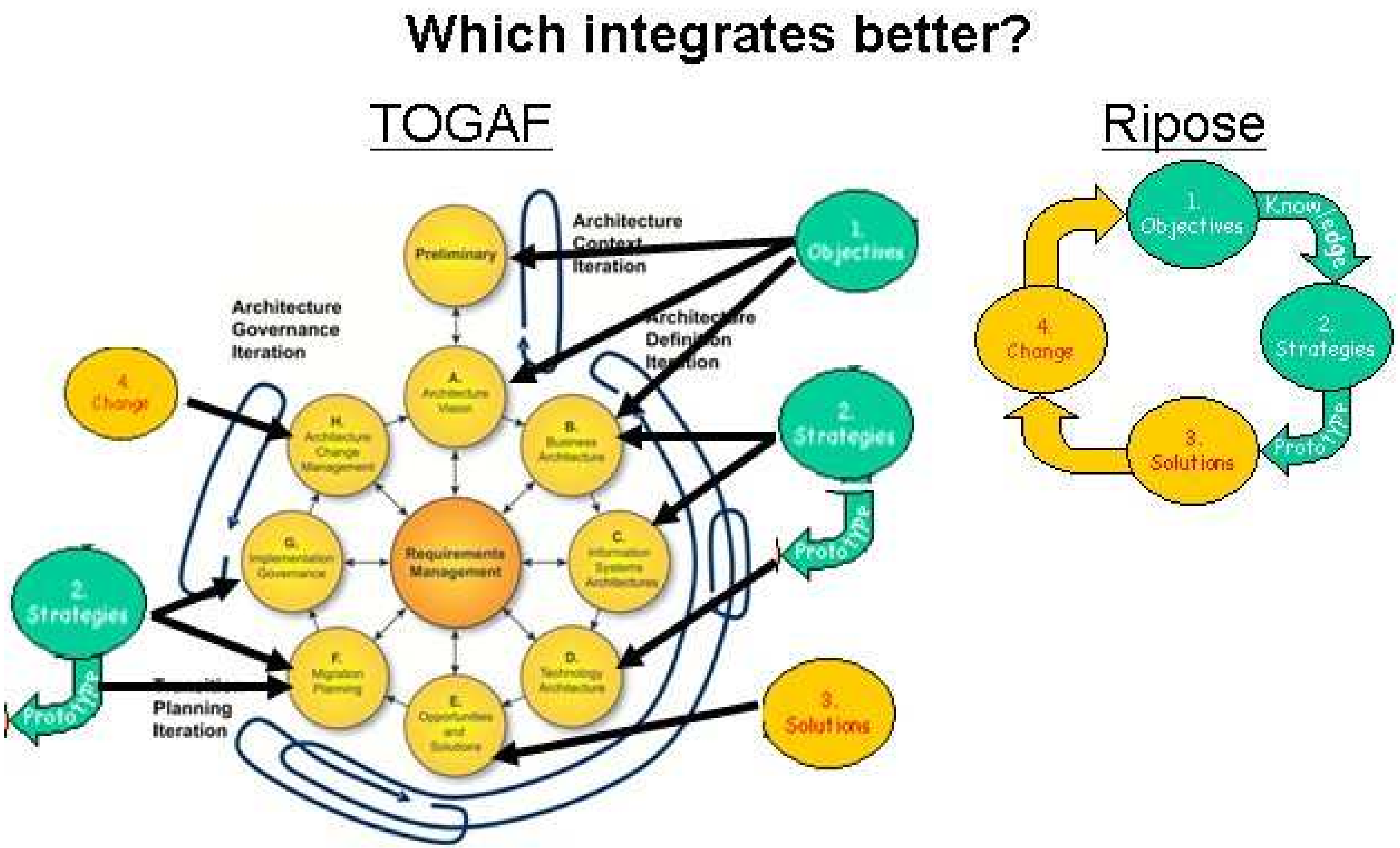
Domain	Step		TOGAF deliverable		State	Ripose Step	Focus	State
Business	H	Change management	Too many		Implicit	1	Information	Implicit
		Preliminary	*Proof of concept?* Too many			1.1	Concepts	
	A B	Architecture vision Business architecture	Archimate	Catalogues Matrices		1.1.1	Objectives	
				Diagrams		1.1.1.1	Goals	
						1.1.1.1.1	Purpose statement	Explicit
						1.1.1.1.2	Benefits	
						1.1.1.1.3	Values	
						1.1.1.1.4	SWOT	
						1.1.1.2	Measures	Implicit
						1.1.1.2.1	Key performance indicators (KPIs)	Explicit
						1.1.1.2.2	Performance indicators	
						1.1.2	Knowledge - industry specific	
	C	Information systems architecture		Process flow diagrams		1.1.3	Actions	Implicit
	G	Implementation governance				1.1.3.1	Systems	Explicit
						1.1.3.2	Sub-systems Industry dependent	
Technology	D	Technology architecture	*Proof of logic?*			1.2	Logic	Implicit
						1.2.1	Facts	
			Archimate	Manually created models		1.2.1.1	Data	Explicit
						1.2.1.2	Databases	
	E	Opportunities & solutions		Use case diagrams		1.2.1.3	Projects	
						1.2.2	Applications	
	F	Migration planning	*Proof of physical?*			1.3	Physical – platform dependent	Implicit
						1.3.1	Database definitions	Explicit
						1.3.2	Programs	

Warnings:

1) Every deliverable is implicit – See The Open Group web site
2) Initial step was incorrect – which one?
3) Models are implicit
4) The remaining steps are in an illogical and silo like (unintegrated) sequence
5) Costs exceed benefits

Compare TOGAF with Zachman

Diagrammatic view



According to my empirical research, TOGAF was developed in 1995 based on the USA Department of Defence’s Technical Architecture Framework for Information Management (TAFIM) which was developed in the early 1990s as a reference model for enterprise architecture. TAFIM starts with ‘Data Architecture’, which, according to my research, was the incorrect starting position. The developers of TOGAF tried to overcome this deficiency but too have failed to come up with a better approach. The loops indicate the number of possible redundant steps.

In 2017 (22 years after the original development) The Open Group introduced the concept of “A new body of knowledge” which in my opinion is too little too late!

Conclusion
Until the Open Group Consortium publishes explicit deliverables: Why bother with this ineffective and inefficient approach?

Back

2. **The Zachman Framework**

Tabular view

Domain	Step		Reification	Zachman deliverable	State	Ripose Step	Focus	State
Business	6	The Enterprise	All 6	*?*	Implicit	1	Information	Implicit
	1,1	Scope content	Identification	*Proof of concept?*		1.1	Concepts	
	1,2	Business Concepts	Definition	Business entity & relationships				
	1,1	Scope Concepts	Identification	Lists of the 6 types		1.1.1	Objectives	
			All 6?	Various models		1.1.1.1	Goals	
						1.1.1.1.1	Purpose statement	Explicit
						1.1.1.1.2	Benefits	
						1.1.1.1.3	Values	
						1.1.1.1.4	SWOT	
						1.1.1.2	Measures	Implicit
						1.1.1.2.1	Key performance indicators (KPIs)	Explicit
						1.1.1.2.2	Performance indicators	
	1,2	Business Concepts (?)	?	Repeat of row 3 perhaps?		1.1.2	Knowledge - industry specific	
	1,3	System Logic	All 6	Various models		1.1.3	Actions	Implicit
						1.1.3.1	Systems	Explicit
						1.1.3.2	Sub-systems Industry dependent	
Technology	1,4	Technology Physics	All 6	*Proof of logic?*		1.2	Logic	Implicit
						1.2.1	Facts	
						1.2.1.1	Data	Explicit
						1.2.1.2	Databases	
						1.2.1.3	Projects	
						1.2.2	Applications	
	1,5	Tool Components		*Proof of physical?*		1.3	Physical – platform dependent	Implicit
						1.3.1	Database definitions	Explicit
						1.3.2	Programs	

Warnings:

1) Every deliverable is implicit

2) Initial step was incorrect

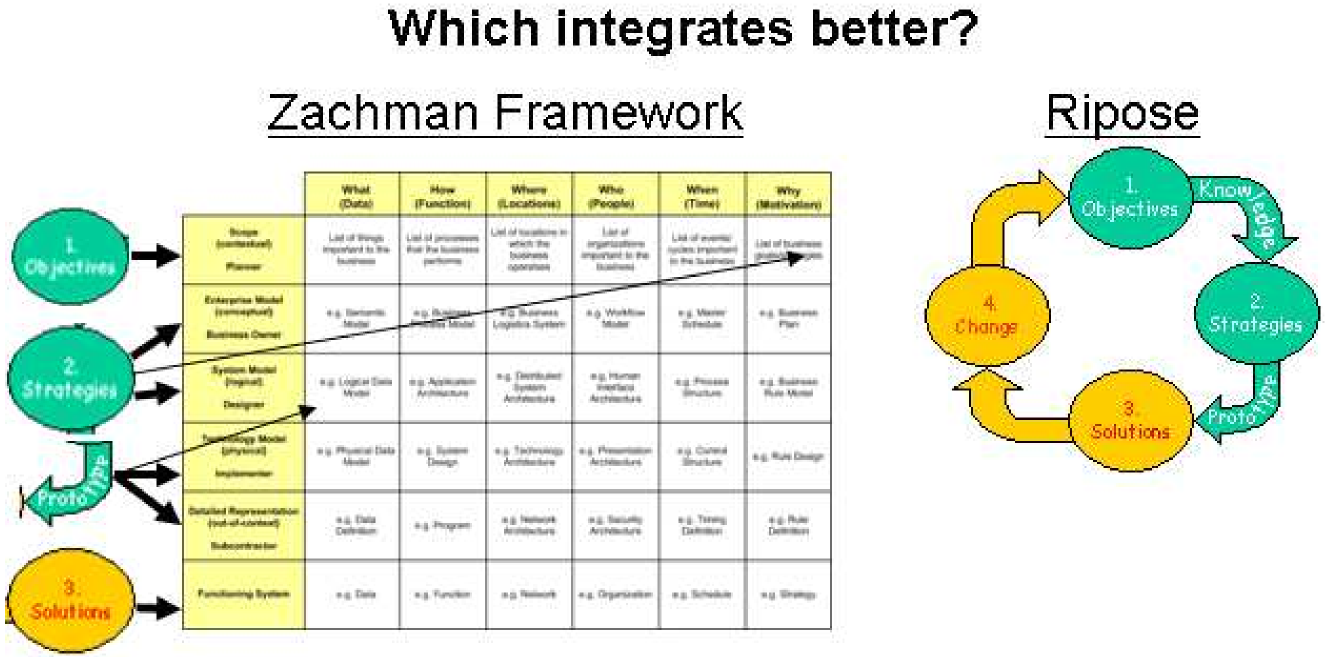
3) Models are implicit

4) The remaining steps are in an illogical and silo like (unintegrated) sequence

5) Costs exceed benefits

Compare Zachman with TOGAF

Diagrammatic view



Which integrates better?

Zachman Framework

Ripose

	What (Data)	How (Function)	Where (Locations)	Who (People)	When (Time)	Why (Motivation)
Scope (contextual) Planner	List of things important to the business	List of processes that the business performs	List of locations in which the business operates	List of organizations important to the business	List of events/cycles important to the business	List of business goals/strategies
Enterprise Model (conceptual) Business Owner	e.g. Semantic Model	e.g. Business Process Model	e.g. Business Logistics System	e.g. Workflow Model	e.g. Master Schedule	e.g. Business Plan
System Model (logical) Designer	e.g. Logical Data Model	e.g. Application Architecture	e.g. Distributed System Architecture	e.g. Human Interface Architecture	e.g. Process Structure	e.g. Business Rule Model
Technology Model (physical) Implementer	e.g. Physical Data Model	e.g. System Design	e.g. Technology Architecture	e.g. Presentation Architecture	e.g. Control Structure	e.g. Rule Design
Detailed Representation (out-of-context) Subcontractor	e.g. Data Definition	e.g. Program	e.g. Network Architecture	e.g. Security Architecture	e.g. Timing Definition	e.g. Rule Definition
Functioning System	e.g. Data	e.g. Function	e.g. Network	e.g. Organization	e.g. Schedule	e.g. Strategy

According to my empirical research, The Zachman Framework was developed in the 1990s by John A Zachman as

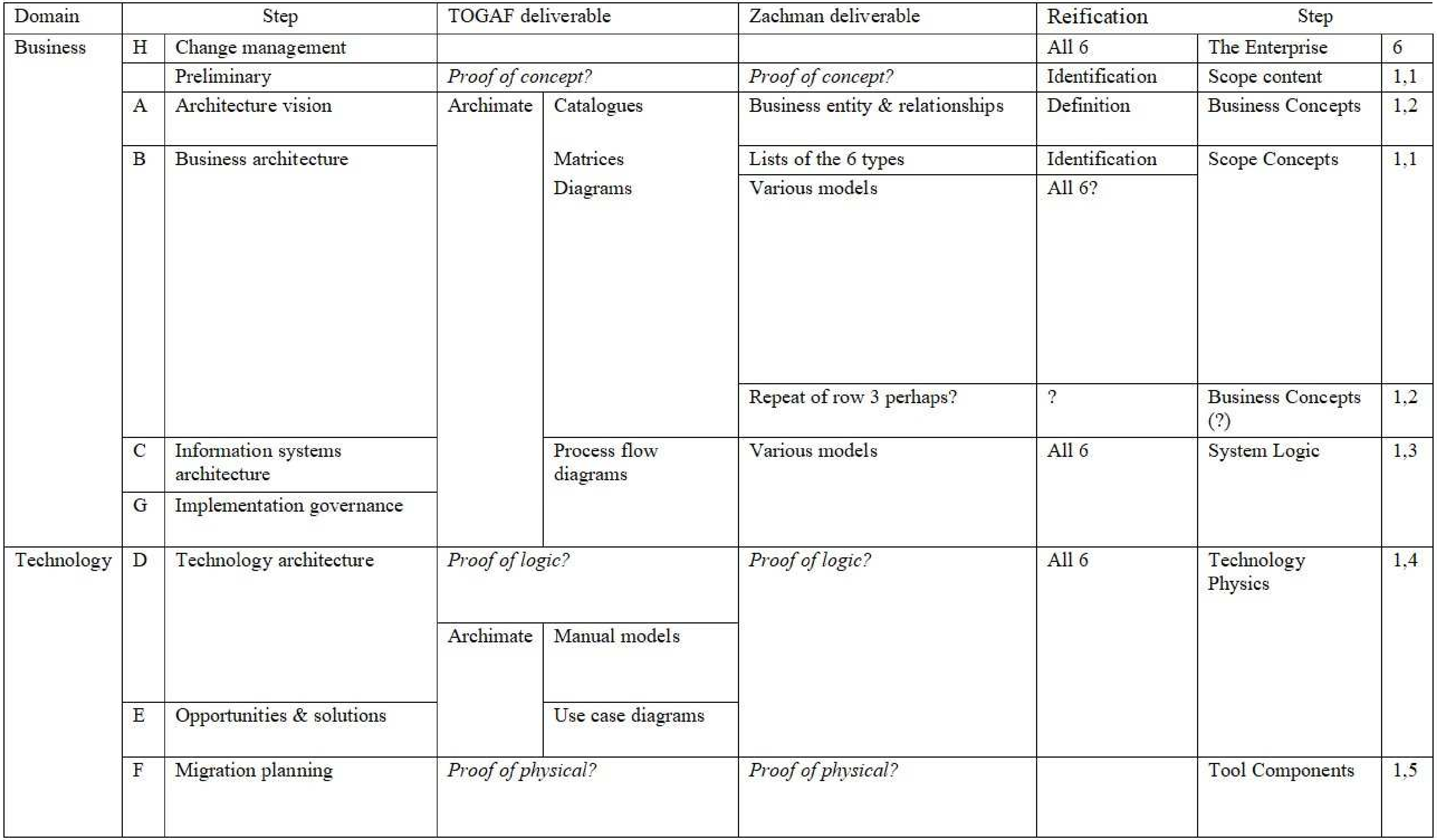
> "***a schema*** *- the intersection between two historical classifications that have been in use for literally thousands of years. The first is the fundamentals of communication found in the primitive interrogatives: What, How, When, Who, Where, and Why. It is the integration of answers to these questions that enables the comprehensive, composite description of complex ideas. The second is derived from reification, the transformation of an abstract idea into an instantiation that was initially postulated by ancient Greek philosophers and is labeled in the* Zachman Framework™*: Identification, Definition, Representation, Specification, Configuration and Instantiation*".

In 2017 (some 30 years after the original development) The Zachman Foundation introduced the concept of "The Business Agility Manifesto Building for Change" which in my opinion is too little too late!

Conclusion

Until the Zachman Foundation publishes explicit deliverables: Why bother with this ineffective and inefficient approach? Back

Comparison between TOGAF and The Zachman Framework



Domain	Step		TOGAF deliverable		Zachman deliverable	Reification	Step	
Business	H	Change management				All 6	The Enterprise	6
		Preliminary	*Proof of concept?*		*Proof of concept?*	Identification	Scope content	1,1
	A	Architecture vision	Archimate	Catalogues	Business entity & relationships	Definition	Business Concepts	1,2
	B	Business architecture		Matrices	Lists of the 6 types	Identification	Scope Concepts	1,1
				Diagrams	Various models	All 6?		
					Repeat of row 3 perhaps?	?	Business Concepts (?)	1,2
	C	Information systems architecture		Process flow diagrams	Various models	All 6	System Logic	1,3
	G	Implementation governance						
Technology	D	Technology architecture	*Proof of logic?*		*Proof of logic?*	All 6	Technology Physics	1,4
			Archimate	Manual models				
	E	Opportunities & solutions		Use case diagrams				
	F	Migration planning	*Proof of physical?*		*Proof of physical?*		Tool Components	1,5

Notes

1) Neither have explicit deliverables

2) Both have different starting points

3) Neither have explicit models. However, Archimate can be used to create the diagrams. Archimate needs to be tailored differently for each approach

4) None of the steps are interchangeable

5) Cost of training in either approach, licensing and tailoring of Archimate

6) See comparison of TOGAF and Zachman with Ripose for a real comparison as to the shortcomings

Back

3. Design Thinking - **The Double Diamond approach**

Tabular view

Domain	Step		Action	Double diamond deliverable	State	Ripose Step	Focus	State
Business	1	Don’t know; Could be	None	?	Implicit	1	Information	Implicit
	1.1	Discover	Research	Client brief - *Proof of concept?*		1.1	Concepts	
				Defined research areas & methods				
	1.2	Define	Build	Themes & clusters		1.1.1	Objectives	
				Insights		1.1.1.1	Goals	
						1.1.1.1.1	Purpose statement	Explicit
						1.1.1.1.2	Benefits	
						1.1.1.1.3	Values	
						1.1.1.1.4	SWOT	
						1.1.1.2	Measures	Implicit
						1.1.1.2.1	Key performance indicators (KPIs)	Explicit
						1.1.1.2.2	Performance indicators	
Technology	1.3	Develop	Ideate	Designs		1.1.2	Knowledge - industry specific	
Business	1.2	Define	Build	Opportunity areas		1.1.3	Actions	Implicit
						1.1.3.1	Systems	Explicit
						1.1.3.2	Sub-systems Industry dependent	
Technology	1.3	Develop	Ideate	Designs - *Proof of logic?*		1.2	Logic	Implicit
						1.2.1	Facts	
						1.2.1.1	Data	Explicit
						1.2.1.2	Databases	
						1.2.1.3	Projects	
						1.2.2	Applications	
	1.4	Deliver	Prototype	Prototypes - *Proof of physical?*		1.3	Physical – platform dependent	Implicit
		Do know; Should be				1.3.1	Database definitions	Explicit
						1.3.2	Programs	

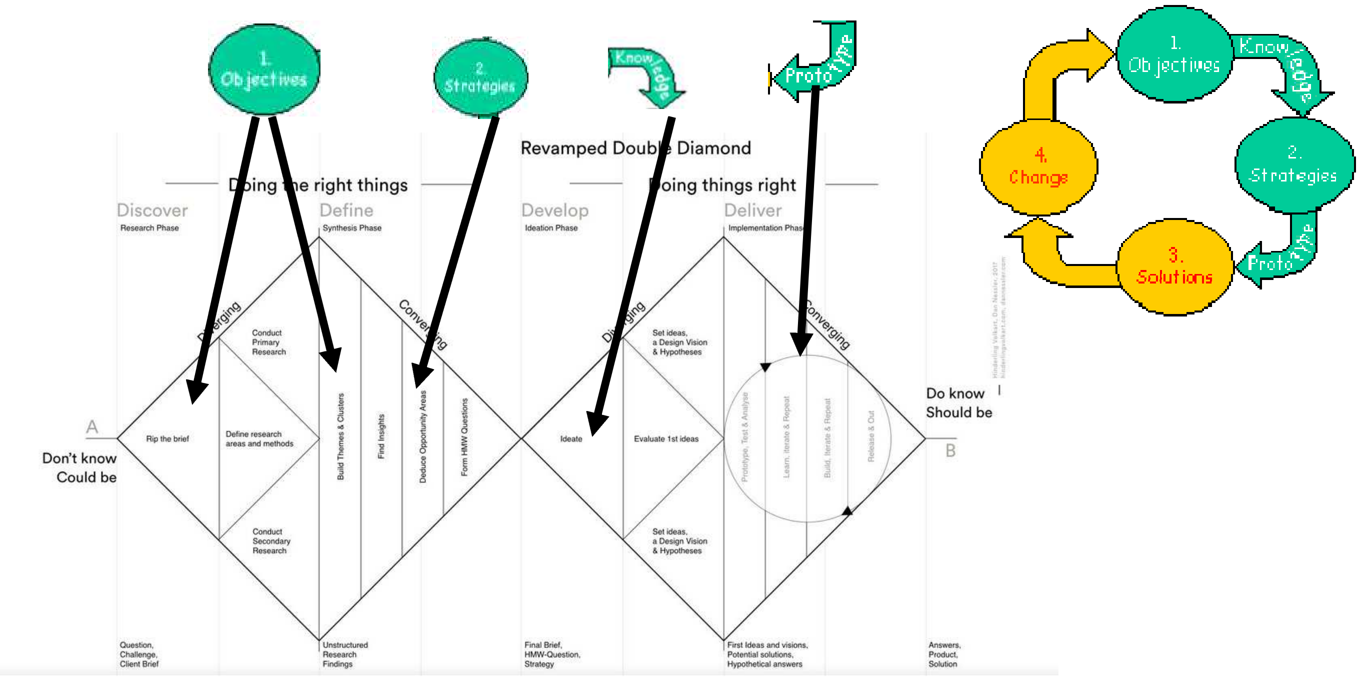
Warnings:

1) Every deliverable is implicit
2) Initial step was incorrect
3) Models are implicit
4) The remaining steps are in an illogical and silo like (unintegrated) sequence
5) Costs exceed benefits

Diagrammatic view

Which integrates better?

Double Diamond   Ripose



According to my empirical research, the DD was developed in the late 2016 by Dan Nessler to demonstrate his approach of “How to solve problems applying a Design Thinking, UX, HCD or any Creative Process from scratch”.

It uses the ‘ideate’ approach (Design thinking) which is similar to the approach used by brainstorming to somehow identify the business knowledge needed to drive prototyping. It appears that there is no connection between ‘business knowledge strategies or objectives’.

Conclusion
Until the originators of Design Thinking publishes explicit deliverables: Why bother with an ineffective and inefficient approach like any of the Design Thinking approaches?

Back

4. **Business & Operations model canvas**

Tabular view

Domain		Step	Canvas deliverable	State	Ripose Step	Focus	State
Business	1	Change	People	Implicit	1	Information	Implicit
	2	Strategy model	*Proof of concept?*		1.1	Concepts	
	2.		Objectives		1.1.1	Objectives	
	2.4		Goals		1.1.1.1	Goals	
	2.2		Vision		1.1.1.1.1	Purpose statement	Explicit
	2.1		Mission		1.1.1.1.2	Benefits	
			No match found		1.1.1.1.3	Values	
					1.1.1.1.4	SWOT	
	3		Metrics		1.1.1.2	Measures	Implicit
					1.1.1.2.1	Key performance indicators (KPIs)	Explicit
			No match found		1.1.1.2.2	Performance indicators	
	4	Business Model	Customer; journeys; Products: capabilities		1.1.2	Knowledge - industry specific	
	2.3	Strategy Model	Drivers		1.1.3	Actions	Implicit
					1.1.3.1	Systems	Explicit
					1.1.3.2	Sub-systems Industry dependent	
Technology	5	Operating model	*Proof of logic?*		1.2	Logic	Implicit
	5.2		Data		1.2.1	Facts	
					1.2.1.1	Data	Explicit
			No match found		1.2.1.2	Databases	
	5.1		Processes		1.2.1.3	Projects	
	5.3		Applications		1.2.2	Applications	
			No match found - *Proof of physical*?		1.3	Physical – platform dependent	Implicit
					1.3.1	Database definitions	Explicit
					1.3.2	Programs	

Warnings:

1) Every deliverable is implicit

2) Initial step was incorrect

3) Models are implicit

4) The remaining steps are in an illogical and silo like (unintegrated) sequence

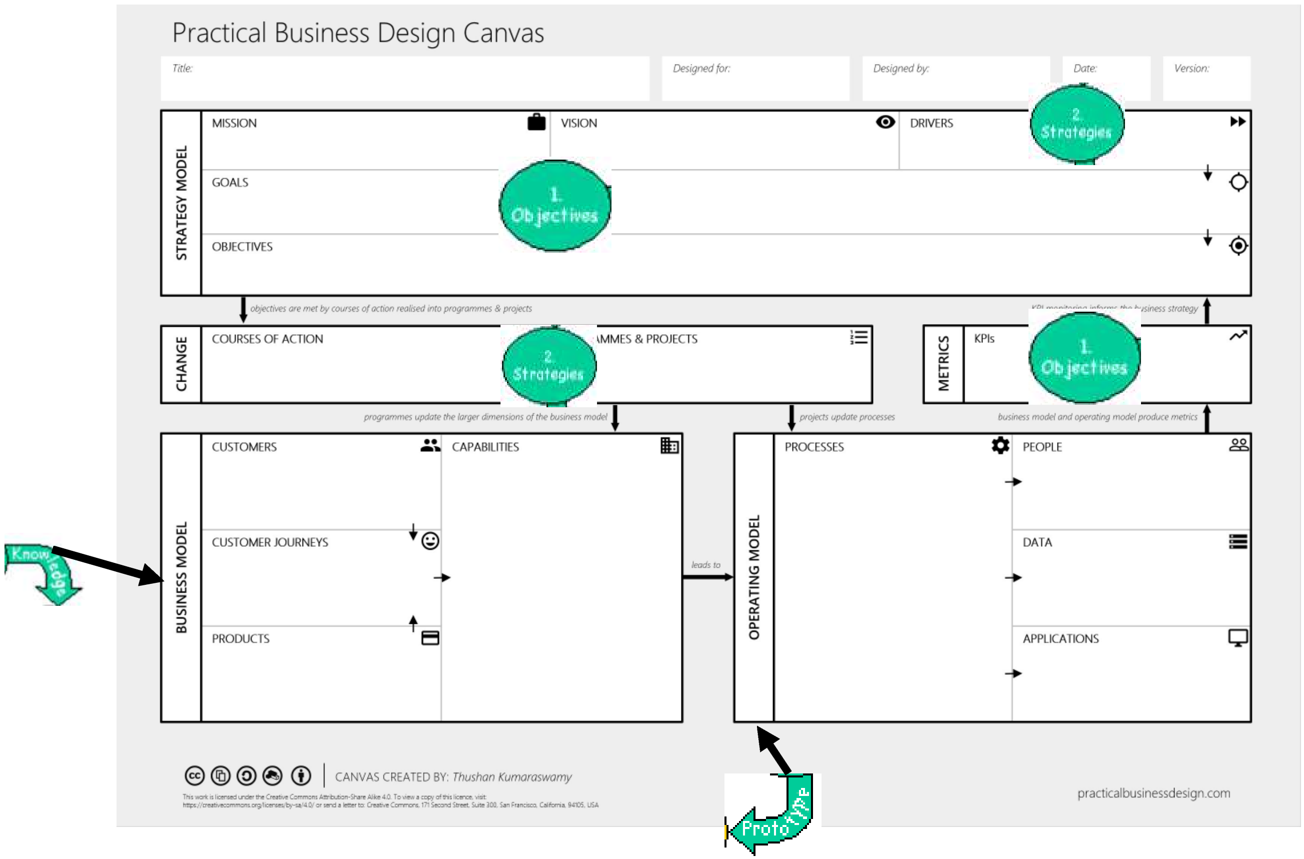
5) Costs exceed benefits

Diagrammatic view

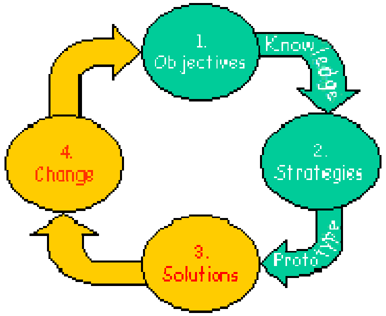
Which integrates better?

Business & Operations model canvas

Ripose



Practical Business Design Canvas
Title:
Designed for:
Designed by:
Date:
Version:
STRATEGY MODEL
MISSION
VISION
DRIVERS
GOALS
OBJECTIVES
1. Objectives
2. Strategies
objectives are met by courses of action realised into programmes & projects
KPI monitoring informs the business strategy
CHANGE
COURSES OF ACTION
PROGRAMMES & PROJECTS
METRICS
KPIs
programmes update the larger dimensions of the business model
projects update processes
business model and operating model produce metrics
BUSINESS MODEL
CUSTOMERS
CAPABILITIES
CUSTOMER JOURNEYS
PRODUCTS
leads to
OPERATING MODEL
PROCESSES
PEOPLE
DATA
APPLICATIONS
Knowledge
Prototype
CANVAS CREATED BY: Thushan Kumaraswamy
This work is licensed under the Creative Commons Attribution-Share Alike 4.0. To view a copy of this licence, visit https://creativecommons.org/licenses/by-sa/4.0/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA
practicalbusinessdesign.com



According to my empirical research, the Business and operations design canvas was developed in 2017 by Thushan Kumaraswamy and at time or writing was still in the development stage to demonstrate his approach to help start-ups and established organisations with Practical Business Designs.

It is based on the Business canvas model as well as the Operating model canvas approaches, each of which have major shortcomings. The original business canvas template provides 9 domains which could generate over 180 business objects.

Conclusion

Until the originators of all canvas approaches publishes explicit deliverables: Why bother with an ineffective and inefficient approach like either the Business canvas model and/or the Operating model canvas approaches?

Back

5. Object Management Group

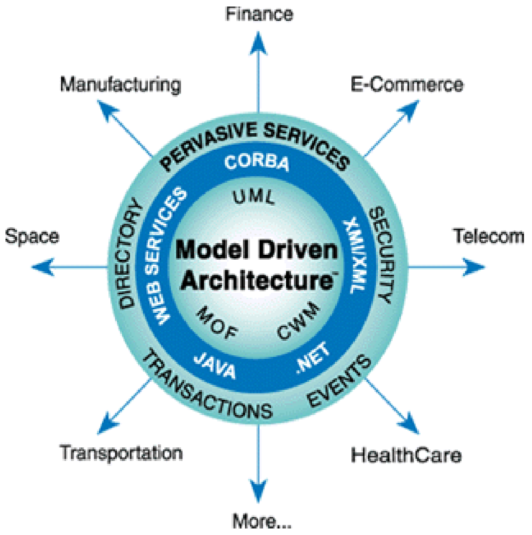
Tabular view

Domain	Step		OMG deliverable	State	Ripose Step	Focus	State
Business			?	Implicit	1	Information	Implicit
	1	Business Process Model & notation	Models of the actual people, places, things, and laws of a domain. The “instances” of these models are “real things”, not representations of those things in an information system. In MDA domain models have historically been called a “CIM” for “Computation Independent Model”. *Proof of concept?*		1.1	Concepts	
					1.1.1	Objectives	
					1.1.1.1	Goals	
					1.1.1.1.1	Purpose statement	Explicit
					1.1.1.1.2	Benefits	
					1.1.1.1.3	Values	
					1.1.1.1.4	SWOT	
					1.1.1.2	Measures	Implicit
					1.1.1.2.1	Key performance indicators (KPIs)	Explicit
					1.1.1.2.2	Performance indicators	
					1.1.2	Knowledge - industry specific	
					1.1.3	Actions	Implicit
					1.1.3.1	Systems	Explicit
					1.1.3.2	Sub-systems Industry dependent	
Technology	2	Case management model & notation	Models of the way the components of a system interact with each other, with people and with organizations to assist an organization or community in achieving its goals. *Proof of Logic?*		1.2	Logic	Implicit
					1.2.1	Facts	
					1.2.1.1	Data	Explicit
					1.2.1.2	Databases	
					1.2.1.3	Projects	
					1.2.2	Applications	
	3	Decision model notation	Models the way in which a particular system or subsystem is implemented such that it carries out its functions. Implementation models are typically tied to a particular implementation technology or platform. *Proof of physical?*		1.3	Physical – platform dependent	Implicit
					1.3.1	Database definitions	Explicit
					1.3.2	Programs	

Warnings:
1) Every deliverable is implicit
2) Initial step was incorrect
3) Models are implicit
4) The remaining steps are in an illogical and silo like (unintegrated) sequence
5) Costs exceed benefits

Diagrammatic view

This is the only diagrammatic representation of the OMG approach, therefore I am unable to map it to the Ripose navigator. However from the information that I managed to obtain from the web, I can only surmise that their deliverables are as implicit as all the other approaches I have researched.



According to my empirical research, the OMG is an international, open membership, not-for-profit technology standards consortium. It was founded in 1989 by eleven companies (including Hewlett-Packard, IBM, Sun Microsystems, Apple Computer, American Airlines and Data General). “The goal was a common portable and interoperable object model with methods and data that work using all types of development environments on all types of platforms” (quotes source).

Conclusion

Until the Object Management Group publishes explicit deliverables: Why bother with this ineffective and inefficient approach? Back

6. **Dual Innovation Management**

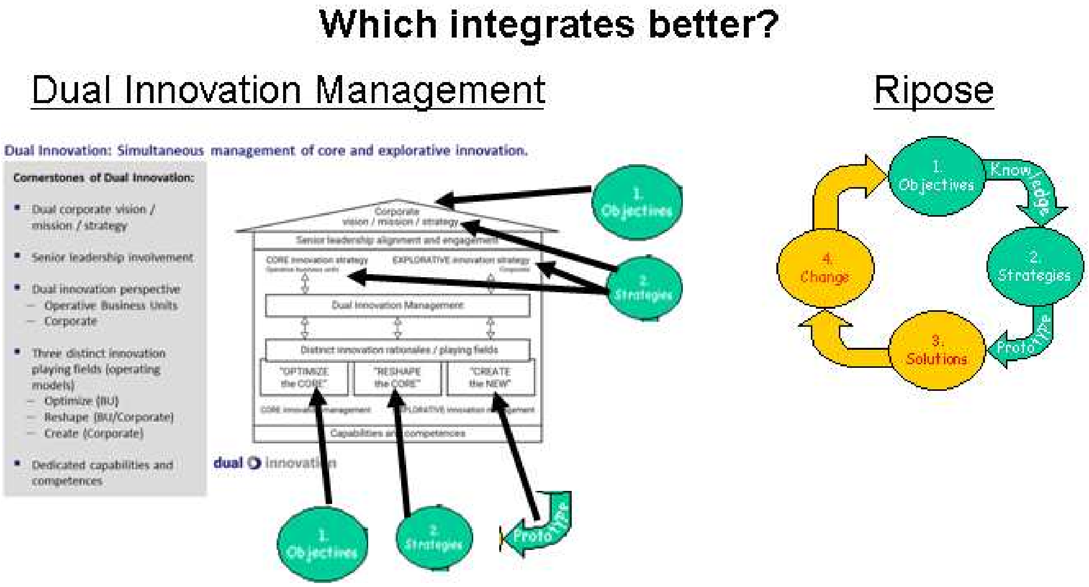
Tabular view

Domain	Step		DIM deliverable	State	Ripose Step	Focus	State
Business		Change management	?	Implicit	1	Information	Implicit
	1	Optimise the core	*Proof of concept?*		1.1	Concepts	
			Objectives		1.1.1	Objectives	
			No match found		1.1.1.1	Goals	
			Vision		1.1.1.1.1	Purpose statement	Explicit
			Mission		1.1.1.1.2	Benefits	
			No match found		1.1.1.1.3	Values	
					1.1.1.1.4	SWOT	
					1.1.1.2	Measures	Implicit
					1.1.1.2.1	Key performance indicators (KPIs)	Explicit
					1.1.1.2.2	Performance indicators	
					1.1.2	Knowledge - industry specific	
	2	Reshape the core	Strategies		1.1.3	Actions	Implicit
					1.1.3.1	Systems	Explicit
					1.1.3.2	Sub-systems Industry dependent	
Technology	3	Create the new	No match found. *Proof of Logic?*		1.2	Logic	Implicit
					1.2.1	Facts	
					1.2.1.1	Data	Explicit
					1.2.1.2	Databases	
					1.2.1.3	Projects	
					1.2.2	Applications	
			Proof of physical?		1.3	Physical – platform dependent	Implicit
					1.3.1	Database definitions	Explicit
					1.3.2	Programs	

Warnings:

1) Every deliverable is implicit
2) Initial step was incorrect
3) Models are implicit
4) The remaining steps are in an illogical and silo like (unintegrated) sequence
5) Costs exceed benefits

Diagrammatic view



Which integrates better?

Dual Innovation Management

Ripose

Dual Innovation: Simultaneous management of core and explorative innovation.

Cornerstones of Dual Innovation:

- Dual corporate vision / mission / strategy
- Senior leadership involvement
- Dual innovation perspective
  - Operative Business Units
  - Corporate
- Three distinct innovation playing fields (operating models)
  - Optimize (BU)
  - Reshape (BU/Corporate)
  - Create (Corporate)
- Dedicated capabilities and competences

Corporate vision / mission / strategy
Senior leadership alignment and engagement
CORE innovation strategy
EXPLORATIVE innovation strategy
Dual Innovation Management
Distinct innovation rationales / playing fields
"OPTIMIZE the CORE"
"RESHAPE the CORE"
"CREATE the NEW"
CORE innovation management
EXPLORATIVE innovation management
Capabilities and competences
dual innovation
1. Objectives
2. Strategies
1. Objectives
2. Strategies
Prototype
1. Objectives
Knowledge
2. Strategies
Prototype
3. Solutions
4. Change

According to my empirical research Innovation management is a combination of the management of innovation processes and change management. It was based on some of the ideas put forth by the Austrian economist Joseph Schumpeter, working during the 1930s, who identified innovation as a significant factor in economic growth. Its aim was to integrate objectives, activities, requirements and inherent tensions along the innovation spectrum as well as enabling aspects, often being discussed independently from each other.

Conclusion
Until the developers of the innovative management idea publishes explicit deliverables Why bother with this ineffective and inefficient approach? Back

7. **Data Modelling**

Tabular view

Domain		Step	DIM deliverable	State	Ripose Step	Focus	State
Business		Your business	Responsive to change(?)	Implicit	1	Information	Implicit
	1	Business model	No match found. *Proof of concept?* Could use any other ‘best practice’ technique and then try integrate the implicit deliverables of them with the implicit models of data modelling		1.1	Concepts	
					1.1.1	Objectives	
					1.1.1.1	Goals	
					1.1.1.1.1	Purpose statement	Explicit
					1.1.1.1.2	Benefits	
					1.1.1.1.3	Values	
					1.1.1.1.4	SWOT	
					1.1.1.2	Measures	Implicit
					1.1.1.2.1	Key performance indicators (KPIs)	Explicit
					1.1.1.2.2	Performance indicators	
	2	Data model	Conceptual data model		1.1.2	Knowledge - industry specific	
			No match found		1.1.3	Actions	Implicit
					1.1.3.1	Systems	Explicit
					1.1.3.2	Sub-systems Industry dependent	
Technology	3	Data model	Logical data model *Proof of logic?*		1.2	Logic	Implicit
					1.2.1	Facts	
					1.2.1.1	Data	Explicit
					1.2.1.2	Databases	
			No match found		1.2.1.3	Projects	
			Pseudo code		1.2.2	Applications	
	4	Implement	Physical databases *Proof of physical?*	Explicit	1.3	Physical – platform dependent	Implicit
					1.3.1	Database definitions	Explicit
			No match found	Implicit	1.3.2	Programs	

Warnings:

1) Every deliverable is implicit
2) Initial step was incorrect
3) Models are implicit
4) The remaining steps are in an illogical and silo like (unintegrated) sequence
5) Costs exceed benefits

Diagrammatic view

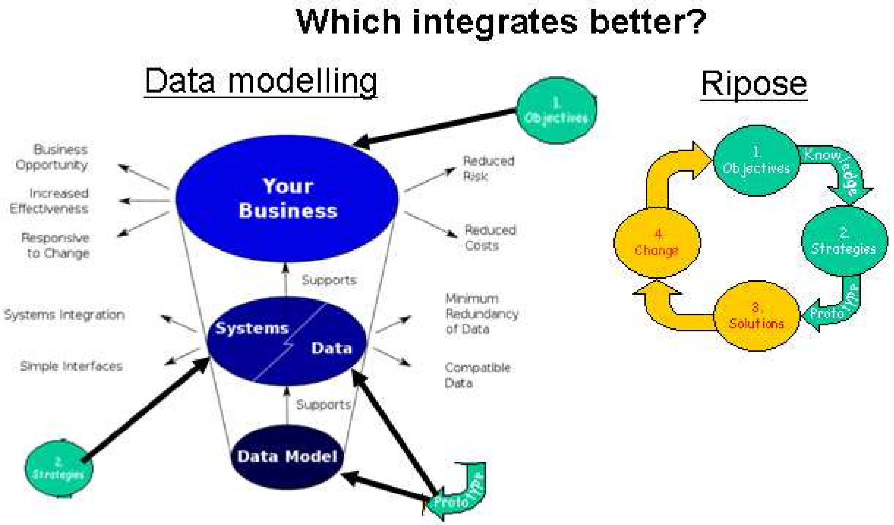


Diagram origin

According to my empirical research “Data modeling in software engineering is the process of creating a data model for an information system by applying certain formal techniques” (see previous page for the link). It is “used to define and analyze data requirements needed to support the business processes within the scope of corresponding information systems in organizations”. It was first introduced between 1960 and 1999, and included “the development of Database Management Systems (DBMS) known as hierarchical, inverted list, network, and during the 1990s, object-oriented Database Management Systems” (see a brief history).

The major problems with this approach are

1) There is no one source of explicit business requirements. Anyone can use any of the ineffective and inefficient approaches (some already mentioned above)

2) The so called ‘conceptual data model’ is an oxymoron. It tries to integrate 2 disparate phases (the conceptual, which lies in the realm of ideas) with that of logic (planning). The danger is that ideas are ‘fuzzy’, whilst logic cannot be

3) The development of the logical database depends on

3.1) Identifying the existence of every possible attribute (aspect, characteristic, construct quality, datum, distinction, element, fact, feature, form, hallmark, idiosyncrasy, indicator, mark, peculiarity, property, quirk, sign, status symbol, sure sign, telltale sign, trademark, trait)

3.2) Removing all redundancies

3.3) Grouping these ‘attributes’ logically to form a network of ‘entities’ that somehow relate to one another through the use of implicit techniques such a normalisation, semantic modelling or even object orientation techniques

Conclusion

Until the originators or experts in data modelling publishes explicit deliverables: Why bother with this ineffective and inefficient approach?

Back

8. **PEAF**

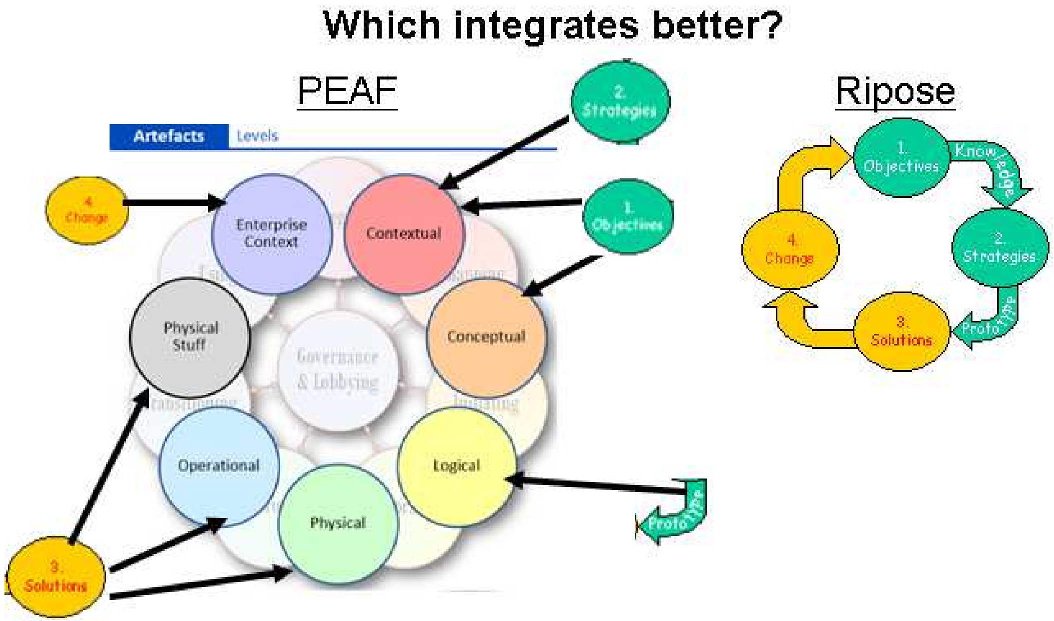
Tabular view

Domain	Step		Focus	PEAF deliverable	State	Ripose Step	Focus	State
Business	1	Using	Physical stuff	?	Implicit	1	Information	Implicit
			Enterprise context					
	3	Roadmapping	Contextual	*Proof of concept?*		1.1	Concepts	
			Conceptual					
	4	Initiating						
						1.1.1	Objectives	
						1.1.1.1	Goals	
						1.1.1.1.1	Purpose statement	Explicit
						1.1.1.1.2	Benefits	
						1.1.1.1.3	Values	
						1.1.1.1.4	SWOT	
						1.1.1.2	Measures	Implicit
						1.1.1.2.1	Key performance indicators (KPIs)	Explicit
						1.1.1.2.2	Performance indicators	
						1.1.2	Knowledge - industry specific	
	2	Strategising	Enterprise context			1.1.3	Actions	Implicit
			Contextual			1.1.3.1	Systems	Explicit
						1.1.3.2	Sub-systems Industry dependent	
Technology	4	Initiating	Logical	*Proof of logic?*		1.2	Logic	Implicit
	5	Elaborating						
						1.2.1	Facts	
						1.2.1.1	Data	Explicit
						1.2.1.2	Databases	
						1.2.1.3	Projects	
						1.2.2	Applications	
			Physical	*Proof of physical?*		1.3	Physical – platform dependent	Implicit
	6	Construction				1.3.1	Database definitions	Explicit
			Operational					
	7	Transitioning				1.3.2	Programs	

Warnings:

1) Every deliverable is implicit
2) Initial step was incorrect
3) Models are implicit
4) The remaining steps are in an illogical and silo like (unintegrated) sequence
5) Costs exceed benefits

Diagrammatic view



According to my empirical research PEAF is “a vendor and consultancy independent, technology neutral, Enterprise Architecture Framework which allows organisations to kick start or re-start an EA initiative and provides a comprehensive set of Products and Processes of everything required to hit the ground running”.

It was first introduced between 2003 and 2008 stating it will be “Cutting EA to the Bone and providing everything you need and nothing you don’t”.

Conclusion
Until the originators of PEAF publishes explicit deliverables: Why bother with this ineffective and inefficient approach? Back