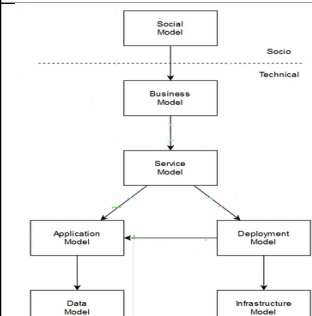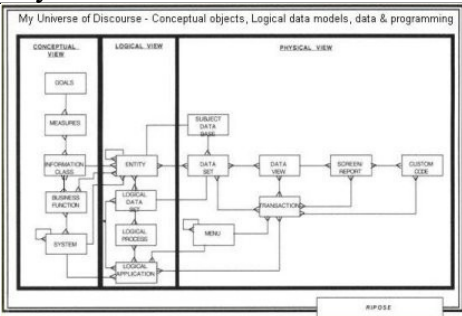6 Oct 2020-10-06
Universes of Discourse

Updated: 6 October 2020 @ 2:23 PM

A comparison between our Universes of Discourse (UoD). I am doing this as I needed to know if your approach would be an improvement on my own or if I could decipher what your approach is capable of. An approach is only as good as the deliverables its practitioners produce.

To this end I decided to compare our UoD and the models that we use in order to gain a better insight into your claims.

Do I care what you do? Not in the slightest. I have at least satisfied myself that yours will need a lot more work if you are to even to try to convince other practitioners (especially TOGAF as The Open Group holds the minds of CIOs around the world.

The UoD

	Yours	Mine
Basis	Social-Technology	Conceptual|Logical|Physical
Developers	Albion Woodbury Small and Charles Richmond Henderson	Platonic Realism from Socrates backed up by Emmanuel Kant
Date	c1898	c388BC & c1788
1st level	Socio	Conceptual
2nd level	Technical	Logical
3rd level	-	Physical
Diagrammatic		

By comparing your approach to mine I found a few similarities however not enough to prove that your approach will actually work under the 3 domains that I developed mine.

On diagnosing your collaboration model I found a number of similarities between our:
1. Meta model
2. Collaborative model
3. Objectives model
4. Business model

I chose my UoD in 1990 whereas I am not sure when you chose yours. However because you chose to follow the 19th Century developers of the Socio-Technical paradigm whereas I chose a BC philosophical paradigm any errors that the developer failed to correct left it up to the experience and research of you and myself. It is my firm opinion that Small and Henderson got their meta model more wrong than did Plato and therein lies the cause of the problem.

Using the information I have now gathered from your web site I have managed to come up with the following more detailed comparison.

Meta model:
This is the highest level model which shows the structure of your approach. It shows the separation between the Social Model and the Technical Model. A separation, as far as I can see, is the root cause of the problem.
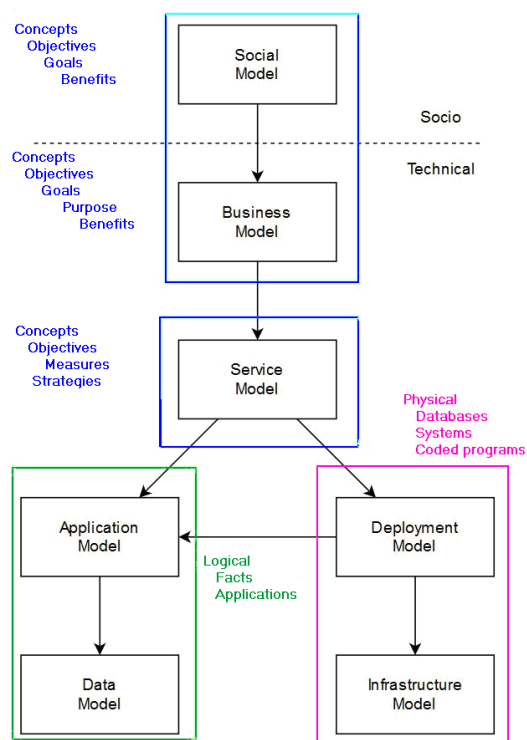
The problems as I see them (just to name 3):
1) Redundancies & conflicts: Caused by the Concepts spread over the 2 universes and 3 domains (Social, Business and Service
2) Disjoint: Caused by the information gathered in the Facts (in the Data Model component) and the Database designs in the Deployment component. These may never be resolved
3) Conflict between the Systems (in the Deployment component) and the Strategies (in the Service component)

Collaborative model

This model is designed to illustrate the deliverable that each model is purported to produce. I decided to overlay these with the deliverable that my approach produces.
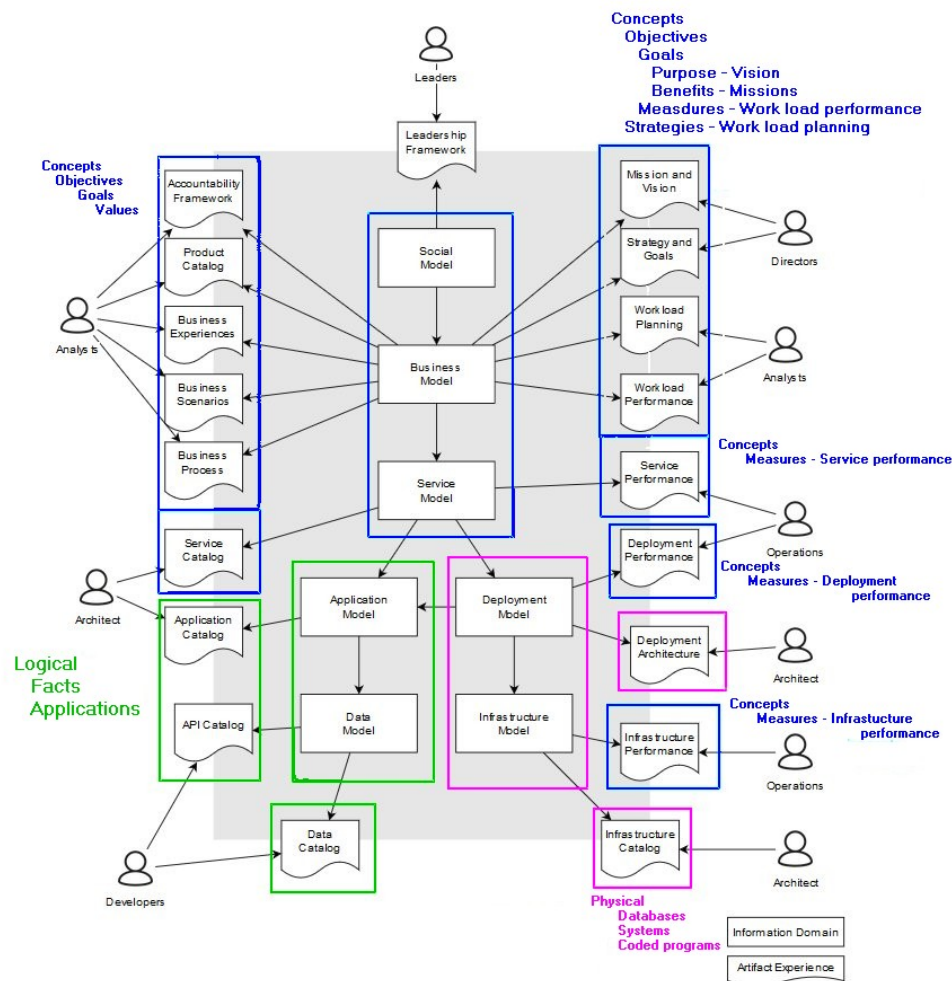
Concepts spread out which could cause redundancies and possible conflicts.

Objectives Meta model

This model is purported to describe how your approach links the delivery of the components of the 'objectives' to other components. In short it is an attempt to show the 'knowledge' components of the approach. The blue rectangles map the to my approach. There are a number of missing components.
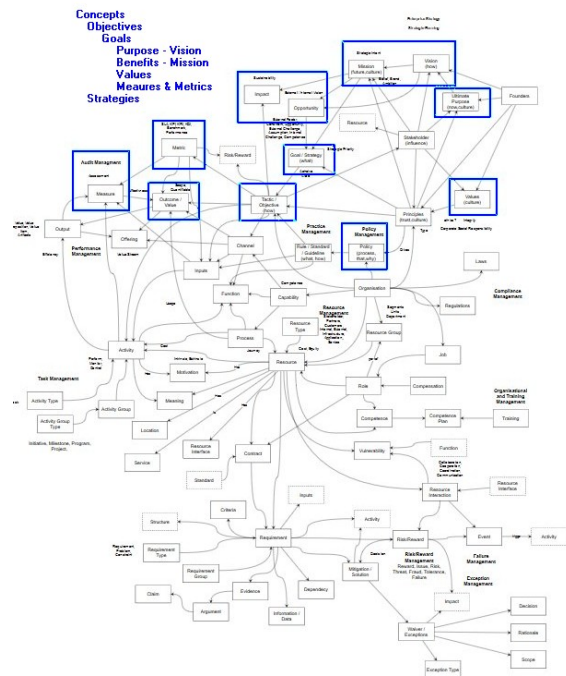
Business meta model

The remainder of the 'objects' try to represent the mapping of what I would call the 'knowledge' objects to the 'objectives'.

The blue rectangles map to my approach. These represent a number of Fundamental and secondary entities but they are not aligned as I would have expected them to be or how I manage them. For example you do not link an 'Offering to a 'Service' and you are missing the 'Product'.
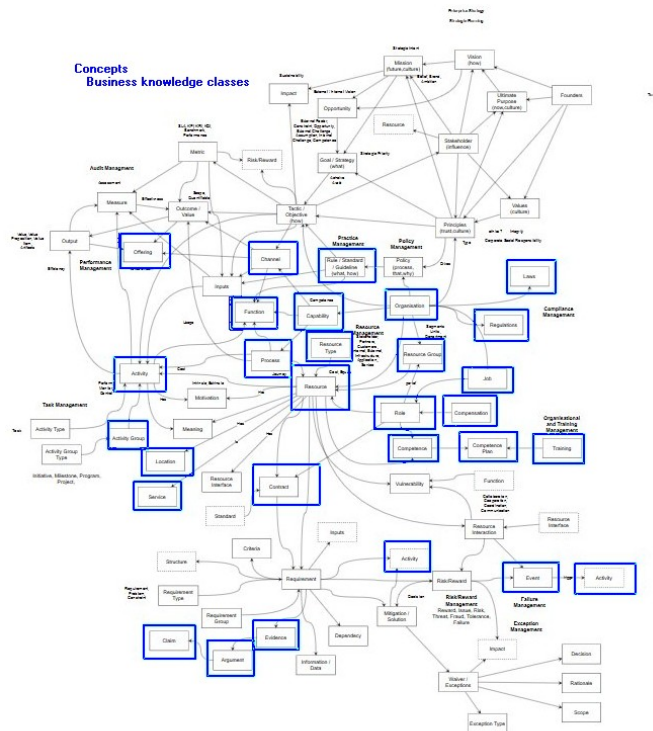
Full mapping with my approach

Using the deliverables and steps that I follow I was now able to show a detailed mapping between our approaches. Perhaps this is where the major differences begin to show.

Domain	UoD	Socio-Technical Deliverable	State	Ripose Step	Ripose deliverable (focuses on)	State
Business			Implicit	1	Information	Implicit
				1.1	Concepts	
	Socio	Social Model		1.1.1	Objectives	
	Technical	Business model		1.1.1.1	Goals	
		Service Model		1.1.1.1.1	Purpose statement	Explicit
				1.1.1.1.2	Benefits	
		Business Model		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 - Strategies	Implicit
				1.1.3.1	Systems - Tactics	Explicit
				1.1.3.2	Sub-systems Industry dependent	
Technology				1.2	Logic	Implicit
				1.2.1	Facts	
				1.2.1.1	Data	Explicit
		Data Model		1.2.1.2	Logical data model	
		Infrastucture		1.2.1.3	Projects	
		Application Model		1.2.2	Applications	
		Deployment & Infrastructure		1.3	Physical – platform dependent	Implicit
				1.3.1	Database definitions	Explicit
				1.3.2	Programs	

Mapping with yours and TOGAF

As I am able to carry out the comparison between my approach and any other I decided to show you the mapping of your approach and TOGAF. His shows me that both approaches have missing components or deliverable that are yet to be explicitly defined.

Domain	UoD	Socio-Technical Deliverable	State		Step	TOGAF deliverables	State
Business			Implicit	H	Change management	Too many	Implicit
					Preliminary	Too many	
	Socio	Social Model		A	Architecture vision	Catalogues	
	Technical	Business model		B	Business architecture	Matrices	
		Service Model				Diagrams	
		Business Model					
				C	Information systems architecture	Process flow diagrams	
				G	Implementation governance		
Technology				D	Technology architecture	Manually created models	
		Data Model					
		Infrastructure		E	Opportunities & solutions	Use case diagrams	
		Application Model					
		Deployment & Infrastructure		F	Migration planning	New systems	

Conclusion
You seem to be on the right path but your selection of the Socio-Technical UoD has constrained your thinking and hence led you to the conclusions you now propose others follow.

As I mentioned it matters not to me what you do with this information as I now know more about your approach so I can only sit back and watch any ensuing argument between you and any of the other Generation X and Generation Y developers who think that all Baby Boomer developers are losers.

Charles Meyer Richter
Principal information architect (RA0)
Ripose Pty Limited
charles.richter@ripose.com