

Conceptual, Logical, Physical: It Is Simple

By

John A. Zachman

Ó Copyright 2000 Zachman International

I don't know why everybody has so much trouble figuring out the difference between Conceptual, Logical and Physical ... let me explain this one more time ...

The "Owner": CONCEPTUALLY "I would like a pot of flowers in the center of my patio about 10 feet off the ground. They would be purely for ascetic reasons, but I want the pot to be BIG and the flowers to be real."

The "Designer": "HmMMM. Let me see now ... the physics of this situation would suggest that there are two LOGICAL alternatives ... either 1) you would have to have a pedestal about 10 feet high, the weight it would have to sustain is max of 100 pounds so if it was 10 square inches in area (cross-section) the material would have to hold 10 lbs per sq. inch. Your second alternative 2) would be to hang it from something above the pot ... do you have a roof over the patio?? If not, that would mean we would have to construct a tripod to suspend the pot from the apex. Do you care if you see the tripod? HmMMM. I recommend you go with the pedestal."

The "Builder:" "HmMMM ... the Architect is suggesting a pedestal that would be 10 feet high and sustain 10 pounds per square inch. That Architect wouldn't recognize a lathe if he fell on one ... but here's what we could do ... we could PHYSICALLY build the thing in three pieces and then glue it together with superglue ... just in case, we could make flanges on the pieces so we could bolt the pieces together to make sure they don't come apart. Your other alternative is to have it made in Japan and ship it in one piece and then we could install it by drilling a hole in the patio, sinking the base down 2 feet and filling in the hole with cement.

Conceptual ... Logical ... Physical. Anyone in manufacturing or construction would recognize the Owner's requirements, the Engineer's design and the Manufacturing Engineer's design ... conceptually what you are trying to produce ... logically how it would be designed ... and physically how you intend to build it.

The information people's problem is, we start at Row 3 and look downwards and we think that Row 3 is Row 2, that "entities" are "Things" ... and then we can't figure out what Row 3 is. Actually, Row 3 is Row 3 ... "entities" are logical representations of Business "Things," not the Business Things themselves. Then the problem is, we never figure out what Row 2 actually is. We start with Row 3. We think that Row 3 is "Requirements" when actually, we never define "Requirements" from an Enterprise

Perspective. We are starting with "Engineering" (logical) design and call it "Requirements." Probably, that is because we are SYSTEMS (Row 3) people, not BUSINESS (Row 2) people.

Row 2 are the models of the actual business as the Owner conceptually thinks the business is or maybe, wants the business to be.

Row 3 are the models of the "systems" of the Business, logical representations of the Business that define the logical implementation of the Business. There are a bunch of "systems." There is the marketing system, the manufacturing system, the distribution system, the management system, the accounting system, the planning system, engineering system, the inventory management system, the check processing system, the personnel system, the management development system, the legal system, the library system, and so on, and so on and, the information system.

There are two reasons why the logical models (Row 3) are so important to the Enterprise. First, you can't design something (Enterprise included) using the actual material components of the product (Enterprise). For example, you don't design a hundred story building by trial and error, starting with logs, I-beams, steel cables, a tank of plastic and some welding rods. No. You start with the logical representation of all of those things on a piece of paper ... the Engineering Drawings, etc. That's the only way you will be able to think through the detail and make sure the laws of physics are obeyed before you actually start constructing the pieces. It is a LOT cheaper to try everything out on paper before you try to fit the actual materials together by trial and error!

From an Enterprise point of view, you wouldn't start with a warehouse full of Raw Materials, an office building full of people and a distribution center full of finished Products and try to form them into a coherent Enterprise. No. You would start with the logical representations of these things and sketch out how it would work in fact on paper. It would be a LOT cheaper that way! Of course, if you already have an Enterprise that does not have any models and then want to change it, you would have to reverse engineer the models out of the implemented Enterprise so you could then re-engineer it into something coherent. It would still be a lot cheaper (and faster) to reverse engineer the models than to try to change it physically by trial and error ... especially if it is large and complex and if the changes are coming faster than you can get them assimilated and you don't have a lot of time to recover from any mistakes!

If you were defining Logical Models only for that portion of the Enterprise that deals with the actual material transformations of the things of the Enterprise, for example, the transformation of raw material into finished goods, or the manipulation and storage of the actual paper orders received, or the handling of cancelled checks, etc., the resultant models would be the engineering for the "blue collar" systems of the business.

However, a second reason you need the Logical Models is because when the business grows so big that management is no longer able to physically maintain contact with the "Things" of the business, they will have to create data "surrogates" of those things in

order to manage the Things. For example, when the business grows so big that management no longer has physical contact with the products that are being produced, they will have to create a data "surrogate"(a "white collar" system) for the products so they can keep track of the products and know what is present and what is absent.

That is, if the Enterprise gets so big and so complicated that when they want to know how many products are ready for shipment, they don't have to go out and physically look for the products, counting them by sight and then deciding what to do ... no, they access their "information system" (the "white collar" system, manual or automated) of data "surrogates" that parallels (or maybe, "mirrors") the physical system which tells them how many products are ready for shipment ... no physical inventory required. That's the only way they could realistically manage a large Enterprise that has lots of Things, scattered all over the place. However ... when the information system doesn't "align" with the "real" Enterprise, physical system, that is, when it doesn't accurately represent what is really happening ... Management will tend to get really upset!!!

I suspect that the "blue collar" system (actual physical, material "Things" of the Enterprise) and the "white collar" system (data surrogates for the physical, material "Things" of the Enterprise), at the logical level, are actually one in the same system, based on the logical "DATA SURROGATES" for the actual Enterprise "Things". In fact, maybe, not only do the data surrogates mirror the physical Things of the Enterprise, but the information processes (Col. 2) are the "mirror" of the material transformations. Maybe the logical distribution of data surrogates and process surrogates (Col 3) mirror the distribution of the material Things and transformations (Processes.) Maybe the Work Flow of the "White Collar" system mirrors the Work Flow of the physical material. Maybe the logical systems cycles mirror the material cycles. Maybe the systems "constraints" mirror the Enterprise motivation. That is, logically, at Row 3, the Blue and White Collar systems may actually be one in the same.

Then, Row 4 is the technology-constrained, physical implementation design of the systems of the business. This is where the systems specialize ... if the technology is lathes, robots, material handling equipment, optical check sorters, etc., etc., the resultant system is going to be a "blue collar," actual material manipulation system. If the technology is pencils, paper and file cabinets, then the resultant system is going to be a "manual" system (probably a manual, "white collar" system). If the technology is stored programming devices and electronic media, then the resultant system is going to be an automated (probably, "white collar," information) system.

All of these systems designs would be derived from the same logical model, the heart of which is the logical representations of the "Things of the Enterprise." That is, there could be more than one physical implementation for the same logical model ... but the physical implementations would have to be kept in synch or every time management needed to know something, they would have to go out and take inventory or physically sight things. (Once again, if the "systems" are not aligned with the Enterprise reality, the probability is, management is going to get really frustrated!)

For Column 1, I am confident that the Row 2 model is a model of the THINGS of the Enterprise ... think "THINGS", do not think "entities" or "objects" or "information areas" or even "terms" ... but "THINGS." Call them what you want, but you'd better think, "Things" or you are going to get confused!! "Things," ... those are what the Enterprise Owners care about CONCEPTUALLY. (Actually, you probably should call them "Things" as well as thinking "Things" so nobody gets confused, including the "Owners," but especially, WE!)

As an aside, they (the Owners) will care about "intersection "Things" in the Conceptual Model as well as the actual "Things"... in fact, where there is an intersection, they will typically create a piece of paper that records the intersection and then put a serial number on that piece of paper (intersection) and keep an inventory of the pieces of paper ... that is, it becomes an Enterprise "Thing" in its own right. (I have recently seen an Enterprise where management did not create such an intersection "Thing" and they couldn't manage that aspect of their Enterprise as a result. It was only when they saw the Column 1, Row 2 "Semantic Model" (the Conceptual Model of the Enterprise "Things") that they discovered what the problem was, a missing intersection "Thing."

For Row 3, think "logical representations" of Enterprise "Things" ... now we are into "entities" or "objects," etc. This model would be a LOGICAL Data Model ... fully normalized ... the basis for the filing system that records information about the actual "Things" so management can keep track of them without having to physically sight them ... the basis for the "white collar" "information system."

Probably, this Logical Data Model is also the basis for engineering the "blue collar" aspects of the Enterprise as well as the "white collar" aspects, because the "entities" in the model are the logical representations, the surrogates of the actual Enterprise "Things." That is what you would need in order to "Engineer" the Enterprise logically ... not the actual things, but a paper (logical) representation so you can think through all of the physics issues to make sure it will work when you get it implemented.

Then, Row 4 ... would be the physical storage design ... how ... depending on what storage device, file cabinet or storage bin ... are you going arrange the storage of the "Things" so you can find the Thing (data surrogate or actual "Thing") again when you need it. (Which device in which location it goes on is a Column 3 issue). In a relational implementation, the Row 4 Model would be the table structure with the keys, foreign keys and so on, the "Data Design," or "PHYSICAL Data model."

Conceptual, Logical, Physical ... Row 2, Row 3, Row 4 ... Simple ... in an absolute sense, simple.

Having discussed the ideas of "Conceptual, Logical and Physical" from the perspective of the Enterprise, let me speculate why those words are so confusing and misunderstood by those of us who come from the Information community. Similar confusion centers around the words "Requirements, Architecture and Design." I think the source of the confusion around both sets of words is the same.

First of all, it depends upon who you are, what you are thinking is “Conceptual,” “Logical,” and “Physical” ... or “Requirements,” “Architecture,” and “Design.”

Whatever Cell you are working on, the Cell above you looks to you like “Conceptual” (or, “Requirements”) and the Cell below looks like “Physical” (or, “Design.”) By default, you either don’t know what to call what you are working on, or you call it “Logical” (or, “Architecture.”)

If you happen to be working at Row 3, to you, Row 2 looks to be “Conceptual” and Row 4 looks to be “Physical.” However, if you are working at Row 4, Row 3 looks to be “Conceptual” and Row 5 looks to be “Physical.” If you are working at Row 5, Row 4 looks to be “Conceptual” and Row 6 looks to be “Physical.” (You can substitute the words “Requirements” for Conceptual and “Design” for Physical in this paragraph for the same effect.)

Conceptual, Logical, Physical ... it’s all RELATIVE. It depends upon who you are. The “Requirements” for any one Cell are coming from the Cell above and the implementation of that Cell is reflected in the Cell below.

Therefore, please notice, “Quality” is a function of the vertical relationship between the Cells of any one Column. For example, where do you find the quality criteria for Column 1, Row 5, the Data Definition? That is, how do you know whether you have the correct product specification for the data elements? Well ... from the Cell above, the Data Design, Row 4. The quality question is, “does the product specification support the intent of the Data Design?”

Where do you find the quality criteria for the Column 1, Row 4, Data Design? That is, how do you know whether you have the correct Physical Data Model (Data Design)? Well ... from the Logical Data Model, Row 3. The quality question is, “does the Physical Data Model support the intent of the Logical Data Model?”

Where do you find the quality criteria for the Column 1, Row 3, Logical Data Model? That is, how do you know whether you have the correct Logical Data Model? Well ... from the Enterprise Semantic Model, Row 2. The quality question is, “does the Logical Data Model support the intent of the Enterprise Semantic structures?”

In a TOTAL sense, how do you ensure that the data in the Database (Row 6) will meet the intent of Management (Row 2)? Well ... you have to keep all of the quality criteria incrementally ALIGNED vertically from Cell to Cell. Total Quality Management is defined as "Producing Products (Row 6) that meet the ‘Requirements’ of the Customer (Row 2)." That is, INCREMENTALLY, does the Cell you are producing meet the "Requirements" of the Cell above, and get successfully “Designed” (implemented) in the Cell below?

To further the confusion, the word "Architecture" tends to be associated with either "Requirements" or "Logical" whereas "Design" tends to be associated with "Physical." This probably stems from the fact that we, the I/S community, are starting at Row 3 and in an absolute, Enterprise sense, don't truly address Row 2, the Conceptual models. As a result of our Row 2 deficiencies, some people would like to change the name of Row 3 to "Architecture" and Row 4 to "Design" and then eliminate all reference to the words, "Conceptual, Logical and Physical." My opinion is, changing any Row names would only add to the confusion. In order to resolve the confusion, we must perceive the Enterprise in its entirety and in an absolute sense perceive:

- Row 2. "The Owner's View."
- Row 3. "The Designer's View."
- Row 4. "The Builder's View."

RELATIVELY speaking, it is confusing because, with the exception of Rows 1 and 6, it is ALL Conceptual, it is ALL Logical and it is ALL Physical ... and it is ALL Requirements,, it is ALL Architecture and it is ALL design ... because it is ALL RELATIVE, depending upon who you are.

I will forever be dogmatic, the Framework ("The Framework for *Enterprise Architecture*") depicts the total set of descriptive representations relevant for describing an Enterprise and therefore, it is all "Architecture," in the sense that it is all descriptive, short of being the actual Enterprise itself. In an absolute sense, Row 2 are "Models of the Business," the "Owner's View;" Row 3 are "Models of the Systems," the "Designer's View;" and Row 4 are "Technology-constrained Models," the "Builder's View." This is a demonstration of why the Framework is a useful communication tool ... to help be "absolutely" specific about what we are talking about so we can avoid any mis-interpretation of what is Conceptual, Logical or Physical ... or what is Requirements, Architecture or Design.

There is one more factor that should be considered ... from a "Scope" standpoint (i.e. project scope, Department Scope, Enterprise-wide Scope, etc., etc.), the Enterprise should be looked at holistically as these models are being produced as well. That is, for example, it probably is NOT a good idea to look at the Marketing System as a stand alone implementation, or the Manufacturing System as a stand-alone system, or the Accounting System as a stand-alone system, etc., etc. because, this will lead to introducing discontinuity and redundancy into the data surrogates of the Enterprise ... that is, it would result in building "stove-pipes" or "Islands of Automation." In this event, by definition, the "White Collar" system will get out of sync with the "Blue Collar" system and the "Owners" are likely to become very unhappy ... it is only a matter of time. The advisable thing to do is to look at the Enterprise holistically, at least at Row 2 first, and then figure out how to break it up into pieces for implementation incrementally, controlling the discontinuity you are building into ANY of the Columns of models ... but this is a subject for a different discussion.

Conceptual, Logical and Physical ... or Requirements, Architecture and Design ... it is simple ... that is, it is simple as long as you recognize their relativity and establish the ENTERPRISE context absolutely.

*John A. Zachman
Zachman International
2222 Foothill Blvd. Suite 337
La Cañada, Ca. 91011
818-244-3763 (Phone and Fax)
johnzachman@compuserve.com*