“Meta” Matters

Richard Martin
Tinwisle Corp
richardm@tinwisle.com

Edward Robertson
Indiana University
edrbtsn@indiana.edu

Abstract Misunderstandings related to “meta” often cause serious problems in systems engineering. This paper attempts to elucidate some of those misunderstandings, particularly as they apply to systems architectures. It examines how standards can provide discipline and clarity, suggesting how standards themselves must be done without falling into similar “meta” traps.

A significant barrier to understanding the place of architecture in systems engineering is a massive confusion concerning levels of abstraction with respect to the systems and enterprises we construct. Our focus is on issues of “meta” – the word itself and how we think about and talk about the various levels of abstraction – in systems engineering. While “meta-ness” is but one of many issues in systems engineering and architecture, it is the least understood and hence mostly likely to cause difficulties.

Substantial progress on the harmonization of International Standards related to the architecture of enterprises and systems is dependent upon a clear understanding of abstract models and “meta” representations.

The impact of confusion about and with meta-levels is especially strong in systems architecture and information systems because both deal with things of the mind and “meta” is entirely a mental construction. System and enterprise modeling is abstraction and “meta” characterizes kinds of abstraction. Hence understanding “meta-ness” clarifies both models and modeling, avoiding the meta-skew which happens when abstraction levels are confounded.

This paper first examines the many conventional (and not-so-conventional) uses of the term “meta-”. It then mentions some diffiulties that arrise when this term is misunderstood or misused. MORE WHEN DONE

The “Meta-” Term

The dictionary gives several definitions of the prefix “meta-”, the relevant one being “more comprehensive, transcending – used with the name of a discipline to designate a new but related discipline designed to deal critically with the original one (meta-mathematics).”[3] A more syntactic definition is “A prefix meaning one level of description higher”.[**??**] In the original Greek, “meta-” meant “behind” or “after”, as in the anatomical term “metacarpal”. The current use arose because Metaphysics came after Physics in

Aristotle's work. (Thus, in being abstracted from physical to cognitive placement, the term "meta-" has itself undergone a meta-transformation.)

"Meta-" is relative (an order relationship), not a property. This is validated by the occurrence of phrases such as "meta-meta-data". [1] When used with a single term, "meta-" is reflective, (but not reflexive, in the sense that "reflexive" is a possible property of an order) with a sense of "aboutness"; "meta-X" means "X about X". Meta-data is data about data, meta-language is language about language, meta-media is journalists writing about journalists,[1] and meta-models are models about models.[2] Due largely to Hofstadter's popular *Gödel, Esher, Bach: the Eternal Golden Braid*[2], the reflexivity of "meta-" has entered the vernancular meaning explicit self-reference and thus is associated with paradoxes from Epimenidies to Russell and the paradox-like proofs of Gödel and Turing.

"Meta-" often has a sense of abstraction. This is related to the fact that reflection often involves abstraction. Thus a meta-model is an abstraction of a family of models.

There are three distinct scales which are sometimes thought of as the "meta-" relationship:

Abstraction:	abstract	↔	concrete
Generalization:	general	↔	particular
Detail:	course	↔	detailed

These three scales are named by one of their extremes; that this name is the top extreme for the first two and is the bottom for the last indicates that we most commonly "look up" Abstraction and Generalization while we "look down" Detail. The first two of these scales are valid expressions of meta-ness while the last is not. In particular, the "meta-" relationship is always many to many, while Detail is always one to many (through decomposition). Contrasting these tree, detail has arbitrary but often many levels with tree-like structure genericty has a few levels with a slim tree structure, while abstraction has fixed levels with a linear

[1] Because "meta-" is directional, it would be convenient to have a term that inverts this direction. Since "meta-" is Greek for "after", we might consider the Greek for "before", which is "pro-" (akin to the Latin "pre"). Unfortunately "pro" already carries way too many connotations.

[2] This even fits with metaphysics (consistent with current use of "metaphysics" but not with the origin of the term) if we recall that physics was once also called natural philosophy, and thus metaphysics is philosophy about philosophy.

or very narrow structure.

- is each of these useful as a purposive dimension in some context?
- is each always a many-to-many relationship?

    it is along Abstraction (the problem of not having multiple inheritance illustrates this)

    it is along Generalization (building-type and building-style are both generalization in the realm of civil architecture)

    certainly it is not along Granularity

  how does time differ across these

  how do decisions differ across these

  how do V&V differ across these constraints are V&V rules that cross meta-levels??

In tree-structred hierarchies. granuarity and scope seem to have an obvious inverse relationships, but these must be carefully considered when intertwined with meta considerations. For example, the decision "all zip codes in HR must be nine digits" is detail-level and small grained in the information model but the implementation of this decision applies widely across the organization.

- as we go up the Generalization hierarchy, there are more ranges; as we go up the detail hierarchy, there are more null values

Moving to a more abstract meta-level increases uncertainty but decreases volatility, or alternatively stated, increasing the meta-level decreases concreteness but increases stability.

Meta-mistakes and "Meta-" Mistakes

As the title of this section indicates, there are two kinds of mistakes involving "Meta-". The first is misunderstanding the sense of the prefix or the application of this sense. The second is to confound applications of the 'meta-steps.[3]

An example of the first kind is mere misuse of the term. [4] For example, a ¡META¿ tag in an HTML document is not a tag about tags but a tag about the document in which it resides – "¡META¿" is really

[3] Note that throughout we attempt to seperate the word from its applications, always quoting the word.

[4] Note our attempt to be careful in this manner, in as much as a " 'meta-' confusion" is confusion related to meta-ness, while a "meta-confusion" would be confusion about confusion. As is often the case, these are distinct ideas but close enough to cause confusion.

an abbreviation for "¡METADATA¿". The variety of meanings in other contexts adds further confusions.

An example of the second kind occurs in land-use planning. Meta-level confusions can be seen in stark contrast in land use planning (and other similar civic processes) because it juxatposes stakeholders with very different understandings of "meta" issues. This is, it justaposes planning board stakeholders, who understand that developing a plan occurs at a different level than quotidian concerns about one land parcel, with citizen stakeholders, who have no appreciation of the subleties of meta-levels. Thus a citizen, not recognizing distinctions related to "meta", may show up at meeting writing master plan to complain about a neighbor's hogs.

STILL WOULD BE NICE TO HAVE WAR STORY

Meta-data

Meta-data is a realm particularly rich in issues of "meta-ness". Thus we focus on this realm in our attempt to understand these issues, with the caution that this is only an exemplar and not necessarily the most important source of meta-ness issues.

Meta-data, as noted above, is data about data. There are different ways in which this "about-ness" may happen: meta-data may be about the occurrence and use of data, about the concepts in data, or about the structure and representation of data. Thus there are three kinds of meta-data. An example of the first kind is the "Dublin Core", the most widely used meta-data standard, which came from the library community and specifies important characteristics of documents, such as creator and publisher. In Geographic Information Systems, meta-data originally recorded the time and means of gathering the data, but sloppy use now characterizes any ancillary data as meta-data. The third kind is most commonly "relational meta-data", the table and attribute names of a relational database. The middle kind is where "meta-" as abstraction and meta as "about-ness" coincide and where meta-data transitions into the data model. These three tiers mirror the ANSI/SPARC hierarchy, as shown in Figure 1, which illustrates the parallels with various meta-data applied to a book.

Meta-models, Architectures, & Frameworks

System architectures are of course about architecting systems, while frameworks are about architecting the architecture of systems. Thus framework is meta to architecture in the context of a realized system. An

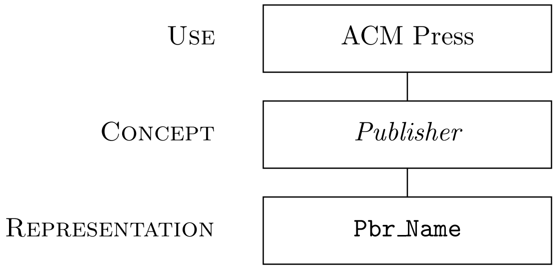

Figure 1. Kinds of Meta-data Parallel ANSI/SPARC Categories

architecture is constructed from models and the meta-models are the major components of the corresponding framework.

Good architecture localizes uncertainty (but does not necessarily limit the consequences) by articulating the relationships between form and function. An architectural framework provides containment for artifacts, and implicitly meta-relationships.

The architecture/design distinction is relative, like up/down, not like top/bottom. This suggests that architecture is therefore "meta-" to design: architecture is certainly over design in scope, and indeed we commonly use terminology such as "Gothic architecture" to characterize certain design patterns and their manifestations. Style is meta along the generalization dimension. An enterprise architecture is the (meta-)design for organizing a design process and capturing design artifacts.

The Entity-Relationship (ER) meta-model is itself expressible as an ER model with only three entities, titled ENTITY, RELNSHIP, and ATTRIBT (Figure 2).

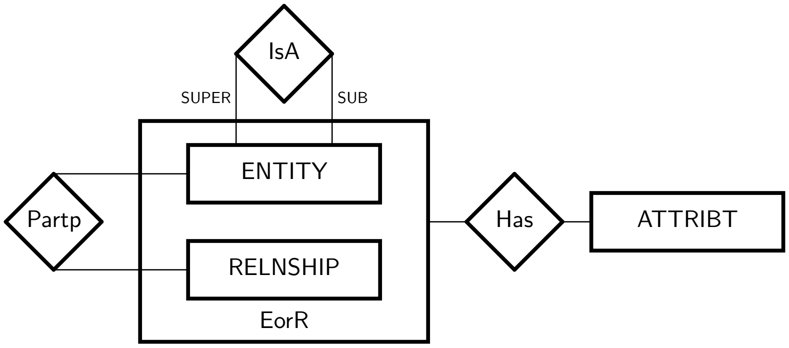

Figure 2. ER Meta-Model for ER Models

One of the strengths of the ER approach is that it can provide its own meta-model.This Figure illustrates the reification induced by a meta-level transition, wherein model-level Relationships map to an Entity (that this Entity is named “RELNSHIP” illustrates the terminological pitfalls that line the “meta-” path). Note that IsA does not reify to a Relationship because IsA is not a first-class construct in an ER model. Such reification may retain the same structural characteristic, which is meta to both a concept and its reification. For example, in Model Driven Architecture, Model Object Facility objects are meta with respect to both Platform Independent Model (PIM) and Platform Specific Model (PSM) levels while the PIM level can be meta with respect to the PSM level.

“Meta-” Confusions

Meta-ness is a very powerful notion, so a mis-understood “meta-” can cause powerful disruptions. This is recognized in a light-hearted manner, as “much hacker humour turns on deliberate confusion between meta-levels,”[??] but where there are jokes there are also serious problems.

“The most common mistake in modeling systems is mixing elements of different levels in the same model,”[??] for example writing a use case at a high level and a creating a class diagram at a low level. A model which mixes meta-levels, where meta-model characteristics are added to a model, is a major source of confusion because this also mixes internals and externals.

The confusion between “specialize” and “instantiate” is often a major hurdle in modeling. This confusion is cleared up by examining the meta-levels involved: specialize is same-level of abstraction while instantiate is down-level to more concrete. Applying these notions to an example involving the PERSON entity:

PERSON $\leadsto$	EMPLOYEE	(specialization)
PERSON $\leadsto$	John Smith	(instantiation)

Transitions to meta-levels do not necessarily go in parallel. The most striking (and problematic) example here is that an information model may model both data and meta-data, but the presence of the that meta-data does not make such a model a meta-model. The above-mentioned parallel between the architecture → framework and model → meta-model is a rare exception where two meta-steps coincide.

people confuse understanding the solution with understanding the problem ’the solution’ often involves one local fix under assumption that context is immutable

unfortunately people get paid for providing a solution

abstractions in the solution space often get confused with abstractions in problem space e.g. - push to characterize all problems in terms of objects

if start solving at low level, need to make assumptions. These assumption may push up detail and meta and get built into solution =¿ make exceptions the rule

constraints @ meta should not constrain solution e.g. to contrary: choice of PL/DE makes certain solutions harder or easier may be necessary to say "must implement in C because ..." but the impacts must also be acknowledged

feedback from instance to meta "roundtrip"

Views & Meta-ness

[[how do viewpoints fit??]]

The notion of View is very general and is "meta-" (in the abstraction sense) to any model, standard, or framework. Views are an essential feature of standards. Most problems with views and meta-ness in standards occur when view considerations incorrectly span meta-levels. Views commonly appear in standards corresponding to[5] a projection along some dimension of the underlying framework. (*e.g.,* an *information view* is a projection along the WHAT interrogative in a Zachman framework)

but calling one dimension "View" ties this top-level "meta-" characteristic into a lower level.

A view expressed at a particular meta-level projects down to lower meta-levels. Indeed, this is so common a mechanism for expressing views that we do it without being aware that we are doing it. For example, an SQL VIEW is defined on tables but applies to instances.

The meta-levels of views are typically specified as if they are distinct, while the instance levels of these views commonly overlap. For example, in a Zachman frame, a model (meta-level) in the "What" column will indicate the abstraction of an artifact and the "How" column, a process; but the actual (instance-level) process applies to the actual artifact.

On the other hand, a view projected up a meta-level is very rarely informative or even non-trivial. The upward view from any ER model yields the same meta-ER model, as noted above.

One would hope that a view through a Model Phase, say Operation, would be coherent. But it may

[5] The phrase "corresponding to" is deliberate, since standards typically do no have an explicit view mechanism.

not be complete historically. One could argue that requirements is meta- w.r.t. design and it is meta- w.r.t. implementation, etc.

Distinct architectural viewpoints conforming to ISO 42010 specify distinct architectural views for which the content may be drawn from the same architectural models.

When a view crosses meta-levels, view updates become very problematic unless very carefully controlled. In the database world, updating table structure while updating data could cause chaos. The common practice that restricts the ability to perform meta-updates provides the necessary control.

Meta and Time

arch use changes over the course of a project (”lamppost” metaphor) does this match V pattern?

decision and V&V granularity a function of time + meta

contraints need time & meta- parameters e.g. contraint in petrochemical factory that ”reactor must always be at least half full” does not allow for start-up

constraints are easy in a static system, while time induces changes requiring (re-)evaluation of constraints

arch should provide clear decision support as arch transforms into design – note this has an inherent meta-knot

BP establishment vs BP improvement is the same meta level but different time points

process (of urban growth and rural property development) has very long time frame (in US, since at least the initial times of European settlement) while meta-process (development of new land-use plans and related ordenances) comes and goes hence processes approx continuous while meta has major and specific step functions (when planning begins to when new ord. comes into effect).

Towards a Solution

The problems with meta-ness are human problems. These problems occur in the way we conceptualize and describe. Thus any solutions must address the human thought patterns. The first step is to spread awareness and understanding of the issues relating to meta-confusions. We hope that the analysis we have done above contributes directly to this awareness and understanding. The single most important factor is simply the warning: *Watch out for ‘meta-’ issues.* This appears deficient in specificity, but that it is in fact its greatest value. As we have seen, the meta-monster is a chimera, able to assume many different forms.

Warning “watch out for snakes” seems insufficient when someone is subsequently stung by a scorpion.

software maturity model has meta considerations towards top, does the same hold for ISO 9000?

Perhaps the hardest thing in systems analysis is extracting what the users need from what the users say. Often this is because users freely wander across the “meta-” boundary. Thus analyists must take special care with “meta-” issues. However, care with “meta-” cannot resolve non-meta issues, as when Joe wants a user interface that looks more impressive than Sue’s. XXX Even more true for standard authors, since they are meta-analyists.

A more powerful tool is found in standards. Standards can and should enunciate general warnings, but having standards writers heed these warnings is much more effective. Several general notions must be manifested is the standards, including:

- To create coherent models, the “meta-ness” of model content must be consistent for all scales of elaboration.
- The usefulness of models to describe collections of systems is relative to coherence across scales of elaboration.
- Standards target different extents of elaboration, with architecture standards focusing on the modeling domain as much as the application domains.
- The higher the meta-level of a standard, the more attention that standard should pay concerning metaness issues.

For example, the “Genericity” dimension of ISO 15704 reflects a meta-model characteristic, in that a Generic, Partial, or Particular view (that is, a slice through the model space along one coordinate of the “Genericity” dimension) is a complete and coherent framework (in the Zachman sense) with a degree of specificity appropriate to that coordinate. A slice along a coordinate of another dimension, say the Information View within the Model View dimension, does not produce a framework but merely a collection of artifacts with comparable content but differing in specificity. Therefore, a Model View view exhibits internals of a framework or set of frameworks. ISO 19440, which presents a Generic model, reflects a solution to this problem.

A model, once in operation, is fixed and its “metas” are irrelevant to that operation. Thus the Operation

Phase of ISO 19439 contains no Generic or Partial models. But when standards dictate that "metas" must be considered, changes are required to be made at the design level rather than "on-the-fly".

In January 2008, the ISO TC184/SC5/WG1 participated in the 2008 workshop of the International Council on Systems Engineering (INCOSE) with the objective of gathering input for standards revisions. The draft report resulting from this workshop includes statements such as "We must distinguish the life-cycle through which the architecting happens from the life-cycle of the architected system; standards need to address both."[6] The requested distinction is one of "meta"'s.

Standards all have multiple facets of meta-ness. Naturally standards are "meta-" with respect to their target domain. Some standards, like ISO 10303 and ISO 15745, produce an initial meta-model and then provide a mechanism for further customizing meta-models for specific sub-domains. This mechanism is then "meta-" with respect to the other meta-models. We note an emerging trend in such standards toward an XML expression of the particular models that may reduce the effort required to implement the standard by allowing an XML meta-model engine to directly implement compliance in the product. Finally, standards often specify compliance mechanisms which are "meta-" to other aspects of the model. This complexity suggests the need for a thorough "meta-meta" analysis.

Revision drafts of ISO 15704 are addressing naming across meta-levels.

difference between model and view are their metas

model vs meta ISO XXXX claims itself to be "metamodel of architectural description"

Arch standard says that it "will not define X", where X is model representation, consistency criteria, *etc.* However, it should require that an architecture conforming to that standard do so. Issue is placing requirements at right meta-level.

arch stds are themsevles meta, in that they govern how other stds are described and articulated

References

1 F. Barringer. A death is noted in the meta-media family. *New York Times*, June 5, 2000.

2 D. Hofstadter. *Gödel, Escher, Bach: an Eternal Golden Braid.* Basic Books, l979.

3 Webster's dictionairy, 7^{th} edition. online version.

[6] As this paper goes to press, the INCOSE report is still in draft form