REPRINTED FROM IBM SYSTEMS JOURNAL, VOL26, NO 3, 1987; © 1987, 1999

A framework for information systems architecture

by J. A. Zachman

With increasing size and complexity of the implementations of information systems, it is necessary to use some logical construct (or architecture) for defining and controlling the interfaces and the integration of all of the components of the system. This paper defines information systems architecture by creating a descriptive framework from disciplines quite independent of information systems, then by analogy specifies information systems architecture based upon the neutral, objective framework. Also, some preliminary conclusions about the implications of the resultant descriptive framework are drawn. The discussion is limited to architecture and does not include a strategic planning methodology.

The subject of information systems architecture is beginning to receive considerable attention. The increased scope of design and levels of complexity of information systems implementations are forcing the use of some logical construct (or architecture) for defining and controlling the interfaces and the integration of all of the components of the system. Thirty years ago this issue was not at all significant because the technology itself did not provide for either breadth in scope or depth in complexity in information systems. The inherent limitations of the then-available 4K machines, for example, constrained design and necessitated suboptimal approaches for automating a business.

Current technology is rapidly removing both conceptual and financial constraints. It is not hard to speculate about, if not realize, very large, very complex systems implementations, extending in scope and complexity to encompass an entire enterprise. One can readily delineate the merits of the large, complex, enterprise-oriented approaches. Such systems allow flexibility in managing business changes and coherency in the management of business resources. However, there also is merit in the more traditional, smaller, suboptimal systems design approach. Such systems are relatively economical, quickly implemented, and easier to design and manage.

In either case, since the technology permits "distributing" large amounts of computing facilities in small packages to remote locations, some kind of structure (or architecture) is imperative because decentralization without structure is chaos. Therefore, to keep the business from *dis*integrating, the concept of information systems architecture is becoming less an option and more a necessity for establishing some order and control in the investment of information systems resources. The cost involved and the success of the business depending increasingly on its information systems require a disciplined approach to the management of those systems.

On the assumption that an understanding of information systems architecture is important to the development of a disciplined approach, the question that naturally arises is "What, in fact, is information

0018-8670/99/$5.00 © 1999 IBM

systems architecture?" Unfortunately, among the proponents of information systems architecture, there seems to be little consistency in concepts or in specifications of "architecture," to the extent that the words "information systems architecture" are already losing their meaning! Furthermore, it probably is not reasonable to expect reconciliation or commonality of definition to emerge from the professional data processing community itself. The emotional commitment associated with vested interests almost demands a neutral, unbiased, independent source as a prerequisite for any acceptable work in this area.

In any event, it likely will be necessary to develop some kind of framework for rationalizing the various architectural concepts and specifications in order to provide for clarity of professional communication, to allow for improving and integrating development methodologies and tools, and to establish credibility and confidence in the investment of systems resources.

Although information systems architecture is related to strategy, both information strategy and business strategy, this paper deliberately limits itself to architecture and should not be construed as presenting a strategic planning methodology. The development of a business strategy and its linkage to information systems strategies, which ultimately manifest themselves in architectural expression, is an important subject to pursue; but it is quite independent of the subject of this work, which is defining a framework for information systems architecture.

Derivation of the architectural concept

In searching for an objective, independent basis upon which to develop a framework for information systems architecture, it seems only logical to look to the field of classical architecture itself. In so doing, it is possible to learn from the thousand or so years of experience that have been accumulated in that field. Definition of the deliverables, i.e., the work product, of a classical architect can lead to the specification of analogous information systems architectural products and, in so doing, can help to classify our concepts and specifications.

With this objective in mind, that is, discovering the analogous information systems architectural representations, the following is an examination of the classical architect's deliverables produced in the process of constructing a building.[1]

Bubble charts. The first architectural deliverable created by the architect is a conceptual representation, a "bubble chart," which depicts, in gross terms, the size, shape, spatial relationships, and basic intent of the final structure. This bubble chart results from the initial conversations between the architect and prospective owner. A sample of such an initial conversation follows:

"I'd like to build a building."
"What kind of building do you have in mind? Do you plan to sleep in it? Eat in it? Work in it?"
"Well, I'd like to sleep in it."
"Oh, you want to build a house?"
"Yes, I'd like a house."
"How large a house do you have in mind?"
"Well, my lot size is 100 feet by 300 feet."
"Then you want a house about 50 feet by 100 feet?"
"Yes, that's about right."
"How many bedrooms do you need?"
"Well, I have two children, so I'd like three bedrooms."

Note that each question serves to pose a constraint (the lot size) or identify a requirement (the number of bedrooms) in order to establish the "ballpark," or approximate conditions, within which any design will take place. From the above dialogue, the architect can depict what the owner has in mind in the form of a series of "bubbles," each bubble representing a room, its gross size, shape, spatial relationship, etc. (See Figure 1.)

The architect's drawings are a transcription of the owner's perceptual requirements.

The architect prepares this bubble chart for two reasons. First, the prospective owner must express what he or she has in mind that will serve as a foundation or basis for the architect's actual design work. Second, the architect must convince the owner that the owner's desires are understood well enough so that the owner will *pay* for the creative work to follow, and in effect, initiate the project.

Having established a basic understanding with the prospective owner, the architect produces the next set of architectural deliverables, which are called architect's drawings.

Architect's drawings. The architect's drawings are a transcription of the owner's perceptual requirements, a depiction of the final product from the *owner's* perspective.

The drawings include horizontal sections (floor plans), vertical sections (cutaways), and pictorial representations depicting the artistic motif of the final structure. The purpose of these drawings is to enable the owner to relate to them and to agree or disagree: "That is exactly what I had in mind!" or "Make the following modifications."

The drawings can be very detailed; however, they are normally developed only to the level of detail required for the prospective owner to understand and approve the design.

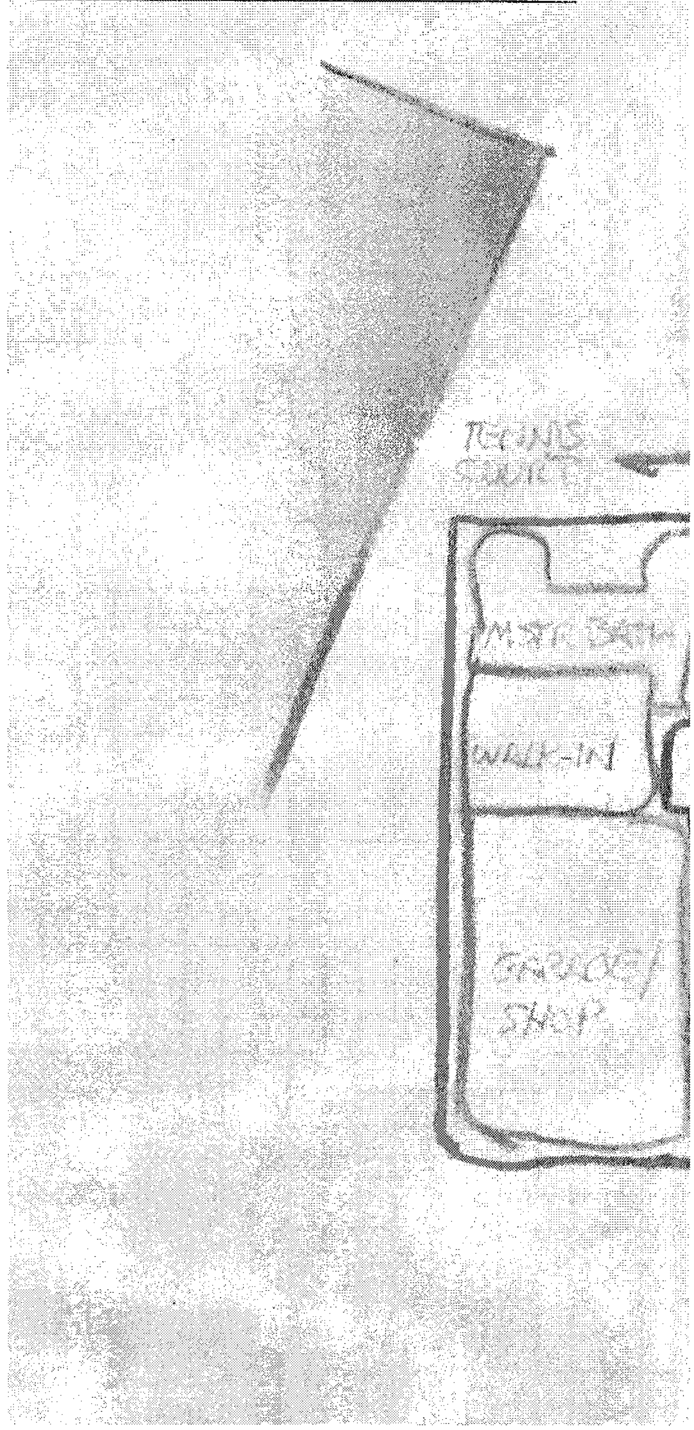
Once the owner agrees that the architect has captured what he or she has in mind, and further agrees to pay the price for continuing the project, the architect produces the next set of architectural deliverables, which are called the architect's plans.

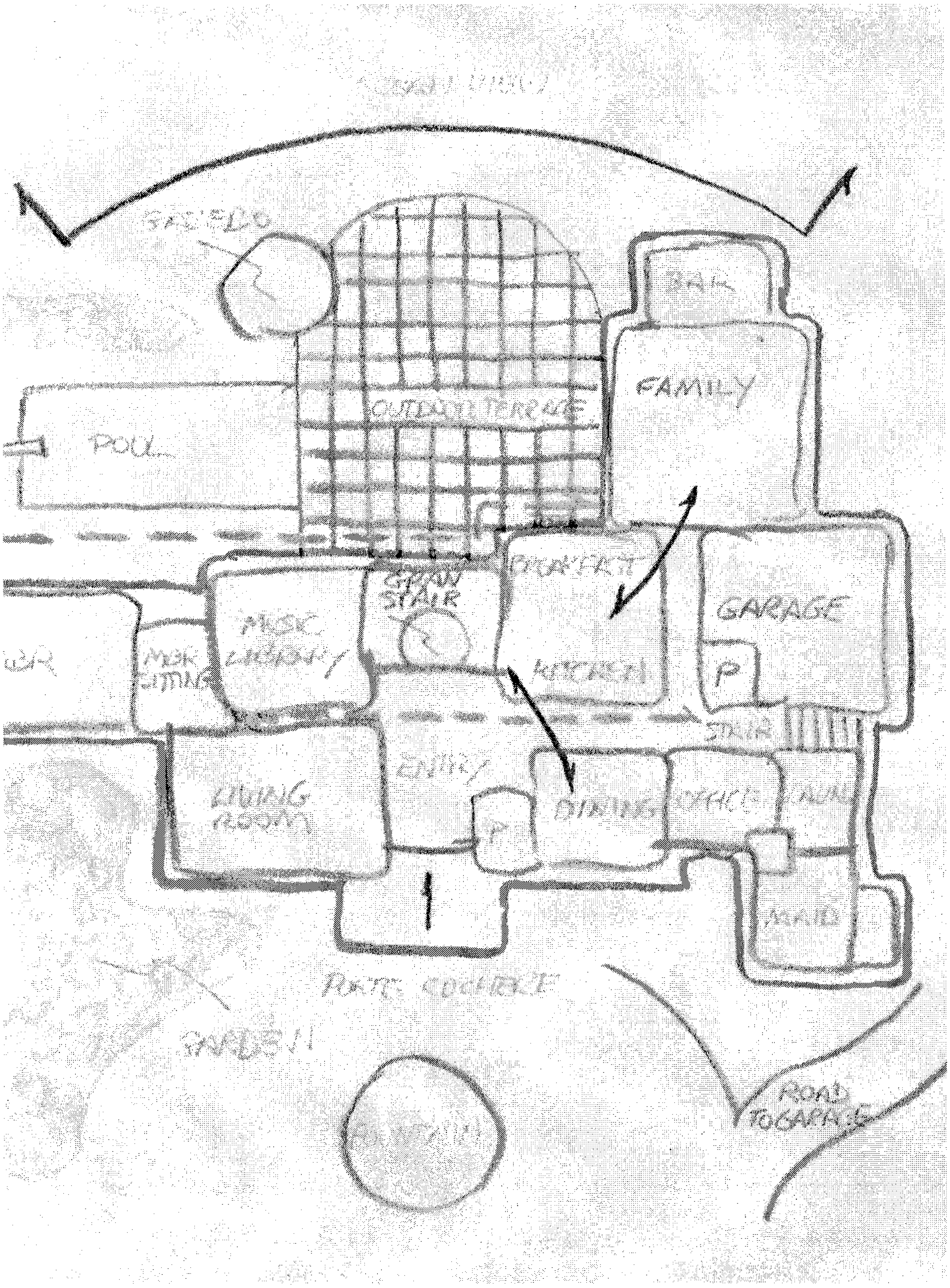
Architect's plans. The architect's plans are the translation of the owner's perceptions/requirements into a product. The plans are a *designer's* representation of the final product (as opposed to an *owner's* representation, which is embodied in the *drawings*). The designer's representation (plans) puts an explicit specification around the material composition of the final product.

The plans are composed of 16 categories of detailed representations, including site-work, electrical system, masonry, wood structure, etc. They describe material relationships in the form of diagrams (drawings) as well as bills-of-materials. These plans are the final deliverables prepared by the architect and ultimately become the official "record" of the finished structure.

The architect's plans are prepared to serve as a basis for negotiation with a general contractor. The owner

Figure 1 Architect's bubble chart[1]




BAR
FAMILY
OUTDOOR TERRACE
POOL
GRAN STAIR
BREAKFAST
GARAGE
P
KITCHEN
STAIR
ENTRY
LIVING ROOM
P
DINING
OFFICE
MAID
PORTE COCHERE
GARDEN
ROAD TO GARAGE

takes the plans to a contractor and says, "Build me one of these." If the contractor builds "one of these," which is represented in the architect's plans, the owner knows that there is a high probability of getting the desired product, as depicted in the architect's drawings.

As a result of the negotiations between the owner and general contractor, the plans may be modified because of cost/price and other considerations, but they finally serve to represent what is committed to construction.

Contractor's plans. At this point, the contractor redraws the architect's plans to produce the contractor's plans representing the *builder's* perspective. Such plans are prepared because complex engineering products are not normally built in a day. Some phased approach is required which, in the case of a building, may comprise first some site work; next the foundation; then the first floor, and so on, until the building is completed. Furthermore, the contractor may have technology constraints. Either the tool technology or the process technology may constrain his ability to produce precisely what the architect has designed. In either case, the contractor will have to design a reasonable facsimile which can be produced and yet satisfies the requirements. These technology constraints, plus the natural constraints requiring phased construction, are reflected in the contractor's plans, which serve to direct the actual construction activity.

Shop plans. Other representations, short of the final structure itself, are prepared by subcontractors. These representations are called shop plans and are drawings of parts or subsections which are an *out-of-context* specification of what actually will be fabricated or assembled. The drawings, architect's plans, and contractor's plans are in-context because the owner, architect, and contractor are all concerned with the entirety of the structure, whereas the subcontractors' representations are concerned with components or parts of the total structure. These shop plans might even serve as patterns for a quantity of identical parts to be fabricated for the project.

The building. In the case of producing a building, the final representation is the physical building itself.

In summary, there is a set of "architectural" representations that are produced during the process of constructing a building. The set is given in Table 1.

Table 1 The set of architectural representations prepared over the process of building a building

Representation	Nature/Purpose
Bubble charts	Basic concepts for building Gross sizing, shape, spatial relationships Architect/owner mutual understanding Initiate project
Architect's drawings	Final building as seen by the owner Floor plans, cutaways, pictures Architect/owner agreement on building Establish contract
Architect's plans	Final building as seen by the designer Translation of owner's view into a product Detailed drawings—16 categories Basis for negotiation with general contractor
Contractor's plans	Final building as seen by the builder Architect's plans constrained by laws of nature and available technology "How to build it" description Directs construction activities
Shop plans	Subcontractor's design of a part/section Detailed stand-alone model Specification of what is to be constructed Pattern
Building	Physical building

A generic set of architectural representations

Now that we have specified the set of architectural representations produced during the process of constructing a building, it becomes apparent that this set of "architectures" may be generic to the process of building any complex engineering product. A cursory examination of military airframe manufacturing appears to validate this hypothesis as follows:

a. *Concepts* equals *"bubble charts"* (ballpark view). The airframe manufacturers begin with some "concepts," which are specifications for the "ballpark" in which they intend to manufacture. For example, concepts for the final product indicating that it will fly so high, so fast, so far, for such and such purpose, with so many people, etc. are formulated to establish its gross size, shape, and performance.
b. *Work breakdown structure* equals *architect's drawings* (owner's view). The work breakdown

structure is the "owner's perspective." The government requires that the manufacturer specify the work to be accomplished in terms of the components/systems against which costs are accrued and schedules are managed. In this fashion, the government controls the manufacturer in the production of the product.

c. *Engineering design* equals *architect's plans* (designer's view). Engineering, the designer, translates the work breakdown structure into a physical product. The resultant "engineering design" is composed of drawings and bills-of-material.
d. *Manufacturing engineering bill-of-materials* equals *contractor's plans* (builder's view). Manufacturing engineering, the builder, applies the laws of nature and technology constraints to the engineering design to describe how to build the product (i.e., inside-out, bottom-up) and to ensure that everything designed is actually producible.
e. *Assembly and fabrication drawings* equals *shop plans* (detail view). Assembly and fabrication drawings are the instructions to the shop floor personnel on how they are to assemble/fabricate the pieces or parts as stand-alone entities.
f. *Machine tool representation* (machine view). Because manufacturing uses computer-controlled equipment to produce some parts, they insert an additional representation of the final piece or part, short of the physical part itself. This representation is a "program" (i.e., "numerical code program") that is a machine language representation.
g. *Airplane* equals *building* (finished product). The final representation is the actual, physical item itself.

An analogous set of architectural representations is likely to be produced in building any complex product.

In any case, there appear to be conceptual equivalents in the manufacturing industry for the architectural representations of the construction industry. This equivalency would strengthen the argument that an analogous set of architectural representations is likely to be produced during the process of building any complex engineering product, including an *information system.*

Before identifying the information systems analogs, it is useful to make some general observations regarding architecture.

First, there appear to be three fundamental architectural representations, one for each "player in the game," that is, the owner, the designer, and the builder. The owner has in mind a product that will serve some purpose. The architect transcribes this perception of a product into the owner's perspective. Next the architect translates this representation into a physical product, the designer's perspective. The builder then applies the constraints of the laws of nature and available technology to make the product producible, which is the builder's perspective.

Preceding these three fundamental representations, a gross representation of size, shape, and scope is created to establish the "ballpark" within which all of the ensuing architectural activities will take place.

Succeeding the three fundamental representations are the detailed, out-of-context representations which technically could be considered architectures because they are representations short of being the final physical product. However, they are somewhat less interesting "architecturally," since they do not depict the final product in total and are more oriented to the actual implementation activities. Nonetheless, they are included in this discussion for the purpose of ensuring a comprehensive framework.

A significant observation regarding these architectural representations is that each has a different *nature* from the others. They are not merely a set of representations, each of which displays a level of detail greater than the previous one. Level of detail is an independent variable, varying *within* any one architectural representation. For example, the designer's representation (i.e., architect's plans) is not merely a succeeding, increasing level of detail of the owner's representation (i.e., architect's drawings). It is different in *nature*, in content, in semantics, and so on, representing a different perspective. The level of detail of the designer's representation (i.e., plans) is variable, and quite independent of the level of detail of the owner's representation (i.e., drawings).