Proceedings of the 2001 Winter Simulation Conference
B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, eds.

A TAXONOMY OF A LIVING MODEL OF THE ENTERPRISE

Larry Whitman
Kartik Ramachandran
Vikram Ketkar

Department of Industrial and Manufacturing Engineering
120G Engineering Building
Wichita State University
Wichita, KS 67260-0035, U.S.A.

ABSTRACT

A designer has a choice of many models, methods, frameworks, and architectures. There is little consistency between these terms among researchers. Some of the most widely used architectures and frameworks are described with definitions and concepts that distinguish them clearly. This paper proposes a clear definition of these terms, a clear distinction between these and a methodology that will significantly aid in the comparison and evaluation of various enterprise models. A direct benefit of this research is a more clear presentation of how the enterprise modeling community uses enterprise models.

1 INTRODUCTION

The paper will first discuss the commonly held views used by enterprise models. The paper then provides consistent definitions of architectures, frameworks and methodologies. The more prevalent architectures and frameworks are then briefly presented. Finally, we conclude with a classification methodology for a living model of the enterprise that will allow for comparison and evaluation of various enterprise models.

1.1 Enterprise Models

A model is an abstract representation of reality. Details that are unnecessary are not included as a rule in most modeling efforts. The modeler determines which aspects of the real system are of interest and which system elements are to be modeled. An additional benefit of Enterprise Modeling lies not with developers or analysts, but with management. Models, requirements, and processes can and should evolve along with the Enterprise - think of it as a living database that you can investigate at any time, examining the processes of a specific part of an enterprise, provide a baseline and use to create a plan for the next project. An enterprise model is defined as "a symbolic representation of the enterprise and the things that it deals with. It contains representations of individual facts, objects, and relationships that occur within the enterprise" (Presley, 1997). The above definition entails the kinds of items that are of interest to the modeler. The use of symbols to represent the enterprise presents these facts, objects, and relationships in an easy to understand manner. A survey was conducted by Whitman and Huff on the use of enterprise models that found that models represented a wide scope, yet 75% did not update their models quarterly (Whitman, 2001b). The typical uses of modeling are (Nathan and Wood, 1991) (Snodgrass, 1993) (Reimann and Sarkis, 1996):

- To analyze and design the enterprise and its processes prior to implementation
- To help reduce complexity
- To communicate a common understanding of the system
- To gain stakeholder buy-in
- To act as a documentation tool for ISO 9000, Total Quality Management, Concurrent Engineering, and other efforts.

1.2 Multiple Views

There are several views of a given enterprise. Different people have different needs for enterprise models. If a single model attempted to contain all data about the enterprise, the model would grow to an unusable state. Therefore, models typically are restricted to representing a single view or perspective of the enterprise. While this promotes understanding by reducing the complexity in the model, it can also lead to disjointed and incomplete understanding of the entire enterprise. The integration of different views is vital to achieving a complete representation of the enterprise.

Previous research defines a number of different views. The Computer Integrated Manufacturing Open Systems Architecture (CIMOSA) work promotes four views: Function, Information, Resource, and Organization (Vernadat, 1992). The Zachman Framework of 1987 (1987) was extended by Sowa in 1992 and describes data (what), function (how), network (where), organization (who), schedule (when), and strategy (why) as the dimensions that must be described. Curtis (1992) defines four views: functional (what process elements are being performed, and what flows of information entities are relevant to these process elements), behavior (when process elements are performed (sequencing)), organizational or resource (where and by whom processes are performed, physical communications mechanisms, storage media and locations), and informational (what information entities produced or manipulated by the process, including data, artifacts, products, and objects). ARIS (Architecture of Integrated Information Systems) also has four views. The three main views used are data, function, and organization. Depending on context (information or business system) the fourth view is either called the resource or control view (Scheer, 1989). Previous work in the development of architectures by the Automation and Robotics Research Institute (Presley et al., 1993) describes a five-view approach:

- Business Rule (or Information) View defines the entities managed by the enterprise and the rules governing their relationships and interactions,
- Activity View defines the functions performed by the enterprise (what is done),
- Business Process View defines a time-sequenced set of processes (how it is done),
- Resource View defines the resources and capabilities managed by the enterprise,
- Organization View describes how the enterprise is organized which includes the set of constraints and rules governing how it manages itself and its processes.

This does not, however, mean that all these views must be present in all models. A model is an abstract representation of reality, which should exclude details of the world, which are not of interest to the modeler, or the ultimate users of the model. Models are developed to answer specific questions about the enterprise. However, multiple views provide a clearer picture of the enterprise and multiple views are useful to answer multiple questions about the enterprise.

1.3 Categories of Processes

Presley et al., (Presley et al., 1993) propose that business processes may be placed into three categories: (1) those processes which transform external constraints into internal constraints (set direction), (2) those processes which acquire and make ready the required resources, and (3) those processes which use resources to produce enterprise results. By providing categories to organize processes, more holistic enterprise designs may be achieved. Figure 1 shows activities (boxes) arranged into business processes (ellipses). The business processes are organized into an enterprise represented by the larger box. At this high level of abstraction, the enterprise itself is represented as an activity that takes inputs and transforms them into outputs using available resources under the bounds of a set of constraints.

Frequently the only activities or processes considered in modeling and improvement activities are those listed as category 3, which transform inputs into products and services. However, it is as important to consider the strategic and acquisition activities in an enterprise. Understanding the different process categories is vital to develop useful representations. Categorizing the different processes helps to ensure that the frequently overlooked categories of setting enterprise direction and acquiring and preparing resources are considered.

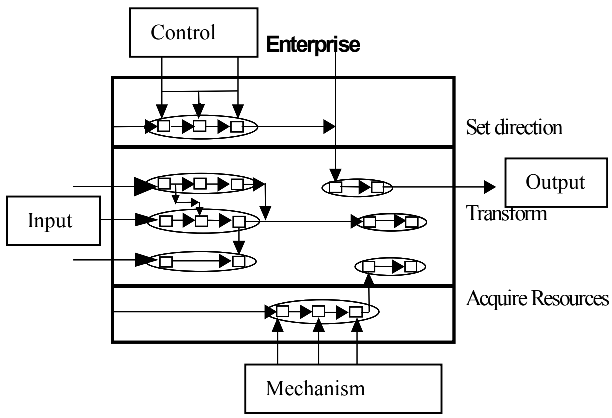

Figure 1: Categories of Processes

2 ARCHITECTURES AND FRAMEWORKS

2.1 Architectures

A system can be formally described by using an architecture. An architecture is made of smaller blocks that define the complete system. Zachman defines an architecture as, "An architecture is that set of design artifacts, or descriptive representations, that are relevant for describing an object such that it can be produced to requirements as well as maintained to the period of its useful life" (Inmon, Zachman, and Geiger, 1997).

An enterprise architecture is fundamental for enabling the assimilation of internal changes in response to the external dynamics and uncertainties of the information age environment. It not only constitutes a baseline for managing changes, but also provides the mechanism by which the reality of the enterprise and its systems can be aligned with management intentions. An architecture is the foundation for managing modern enterprises.

Architectures support a formal reasoning about the present system state. They provide a complete view in which an organization works from the stage of raw material procurement through product building until shipping the final goods to the customer (Inmon, Zachman, and Geiger, 1997). Architectures help in defining the components of the enterprise completely in a way that meets the needs of a business. An enterprise architecture is a tool which can be used for developing a standard method for viewing the system in which the enterprise operates in a very broad range. The most striking advantage is that architectures serve as a common tool for all the employees across the entire enterprise. This in turn helps in enabling the top management of the system to plan where it wants to be as well as develop strategies to get there. Architectures are the building block of an effective strategy, which in turn forms the baseline of any successful modern business strategy.

The salient characteristics of the most widely used architectures are discussed in the following paragraphs.

John Zachman developed the Zachman framework from his experience in various engineering and manufacturing industries. It goes beyond the traditional system development cycle by considering the viewpoints of the various responsible people in the organization. The Zachman framework is essentially a matrix consisting of five rows and six columns. The five rows of the framework represent the individual perspective of five different layers of the organization. The six columns of the framework talk about the various dimensions that an individual perspective has to decide on. By following a sequential methodology as proposed by the Zachman framework an enterprise can approach any project or solve any crisis in a systematic manner. It provides a nomenclature of concepts for relating different dynamics existing in an enterprise. It also provides fundamental distinction to areas overlooked by traditional system design. This framework serves a common language throughout the enterprise to describe the various issues under consideration (Inmon, Zachman, and Geiger, 1997).

CIMOSA (Computer Integrated Manufacturing Open System Architecture) was developed by academic institutions for CIM manufacturers and other government agencies. CIMOSA architectures moves from a general to partial and then to particular case for every view and dimension. It is a process- oriented modeling approach. It promotes descriptive modeling rather than perspective modeling. As compared to the Zachman framework CIMOSA's views and dimensions are developed as per the requirements of the CIM manufacturer (Bernus, Nemes, and Williams, 1996). PERA (Purdue Enterprise Reference Architecture) was developed for enterprise modeling for a CIM (Computer Integrated Manufacturing) factory at Purdue University. PERA establishes a basis for the treatment of human-implemented functions. It represents information system tasks with manufacturing tasks and human-based tasks. PERA clearly defines the extent of automation by distinguishing between humans to those done by the system (Bernus and Nemes, 1996).

The GRAI – GIM methodology was established at the University of Bordeaux to help the designer model a production management system. It is a structured approach supporting the whole life cycle of a system and is divided into two parts: User oriented and technical. GRAI – GIM is used as a reference guide during the implementation or operational phase of a project (Bernus, Nemes, and Williams, 1996).

GERAM (Generalized Enterprise Reference Architecture and Methodology) is a generalization of existing architectures GRAI-GIM, PERA, and CIMOSA made by the International Federation for Information Processing (IFIP) and the International Federation of Accountants (IFAC). This architecture combines the best features of all the existing architectures reviewed by IFIP / IFAC task force (Mills, and Kimura, 1999).

In conclusion, the striking characteristics of any architecture can be briefly summarized in three main points: (a) Taxonomy of concepts for relating things, (b) Clear distinction to areas that were being over looked by traditional System Development Life Cycle which are mainly 1-Strategy, 2 – Analysis, 3 -Design , 4 – Construction, 5 – Documentation and 6 - Production. (c) Common language to describe the subject matter that can be easily understood across the organization.

2.2 Frameworks

The main aim of any framework is to define the relevant business requirements that apply to the evolution of the architecture. "The Framework as it applies to the enterprise is simply a logical structure for classifying and organizing the descriptive representations of an enterprise that are significant to the management of the enterprise as well as to the development of the enterprise's systems" (Inmon, Zachman, and Geiger, 1997). Frameworks define the architectural vision that demonstrate the response to the requirement of the enterprise. The requirements are based on the business goals and strategic resources of the organization. Frameworks define an architecture for a specific purpose and speed up its development.

When architectures are developed to cater for a specific industry or sector they are called a framework. They ensure full coverage of the designed solution. The striking features of the widely used frameworks like IAA (Insurance Application Architecture), EAP (Enterprise Architecture Planning), ISA (Information System Architecture Framework) are described in the following paragraph.

ISA and EAP are adaptations of the Zachman framework, which do not include the people, motivation and time dimensions present in the original framework. ISA is applicable to the Information Systems industry. EAP was developed by the Chief Information Officers (CIO) Council in April 1998 for a federal directive. EAP provides general guidelines to all federal agencies to develop their own information architecture. This promotes Federal interoperability, Agency resource sharing which reduces infrastructural costs for all the agencies as they share data and information. This is mainly used by the departments of the federal government to have consistency while developing a framework (EAP). IAA is a generalized business framework for the insurance industry, developed by collaborations between more than 40 insurers worldwide. It helps to create and maintain a single view of clients, and of the entire enterprise. It also allows designing new business systems infrastructures and has the flexibility needed to react to market dynamics. It can also accommodate changes in corporate strategy (IAA).

In conclusion, the striking characteristics of any framework can be briefly summarized in two main points: (a) It is a plan that helps in achieving the goals of a specific industry, and (b) It is a particular component meant for a particular purpose, meaning that its use is only for a particular purpose.

As can be deduced for the variety of frameworks in the preceding section, there is no single universally correct architecture. We know and we have seen that various architectures have been used to adapt to various kinds of industry to suit various applications.

2.3 Comparison and Contrast

Architectures and frameworks are very commonly used in defining enterprise models. These are terms that are highly inter linked and ambiguous. Due to lack of clarity and understanding of these terms may a time they have been wrongly used in place of the other. This necessitates the need for defining these terms clearly so that they can be clearly understood by all. Hence it becomes essential to understand the demarcation line between these. Our definitions of these two misused terms follow:

Architectures help in building the enterprise system in such a way that it targets the end system while a framework builds on defining the various points of the organization and thereby help in building and defining the architecture. An architecture provides a bigger picture of the entire enterprise by taking into account all possible views, integrates them to provide the bigger picture thereby enabling enterprise to achieve its goals.

A ***framework*** is much more focused as compared to an architecture and is generally used when applied to a particular industry/ situation or sector. The previous section has provided a clear understanding of enterprise architectures and frameworks. It has also provided a clear distinction between the two concepts. With a clear understanding of Enterprise Architectures and Frameworks now Table 1 shown below provides a clear distinction between the two.

Table 1: Comparison between Architectures and Frameworks

Sl. No	Enterprise models	Commonly known as	Should be called as
1	ISA	Architecture	Framework
2	IAA	Architecture	Framework
3	EAP	Architecture	Framework
4	TOGAF	Architecture	Tool for defining an architecture
5	Zachman	Framework	Architecture
6	PERA	Architecture	Architecture
7	GERAM	Architecture	Combination of Architecture
8	GRAI – GIM	Architecture	Architecture

We know that most of the commonly used Architectures and Frameworks have forms that are very similar to a Matrix that have various rows and columns.

A close look at any of Architecture or Framework reveals that they have a few rules that they follow. A few main rules are:

(1)The columns have no order, (2) Each column has a simple, basic model, (3) The basic column of each model must be unique, and (4) Each row represents a distinct, unique perspective. (5) Each cell is unique, (5) The integration of all cell models in one row/column constitutes a complete model from the perspective of that row/ column and they have recursive logic.

The previous sections on architectures and frameworks discussed how an enterprise architect can plan the required models. However, we are interested in understanding the types and uses of existing models within an enterprise. This research is focused upon enabling the reuse of enterprise models. A model must be of value throughout the life of the enterprise to be reused, so it is important to understand the characteristics of useful models. The following section discusses the characteristics of a living model of the enterprise and the subsequent section discusses taxonomy for enterprise models facilitating reuse of these models.

3 A LIVING MODEL OF THE ENTERPRISE

The living enterprise model must have the following characteristics to be effective: (Huff et al., 1991)

Maintainable. A key feature of the model is that the model accurately represents the enterprise at all times. Enterprises change, therefore the model must change. The model must be easily extended to incorporate changes to one aspect of the enterprise, and those changes must be easily incorporated. This leads to the question of a "top-down" or a "bottom-up" approach. The "top-down" approach leads to a more holistic model. The "bottom-up" approach tends to allow for the modeling of an aspect of a system and then connecting the various components as they are validated.

Dynamic. Again, as the system changes, so must the model. Most enterprise models are static. A living model of the enterprise must change as the system changes. It must also provide important information on both the rate of change and the reason for change.

Expandable. The model must also support the addition of new subsystems. Especially in phased implementations, additional aspects of the enterprise will be assimilated into the living model of the enterprise. Therefore, it is imperative the modeling methodology be expandable to include these new models.

Decompositional. Models currently provide multiple levels of detail. This is primarily to provide understanding of the enterprise by various levels of management. The living model of the enterprise must support not only the understanding, but also the decision making and control of the system at various levels of detail.

Consistent with key enterprise metrics. One of the primary goals of a living enterprise model is to ensure that the model has intrinsic value. By creating the model to be consistent with current enterprise metrics, and even creating the model to drive the metrics, the model becomes an integral part of the enterprise.

Driven directly from actual enterprise data. The inputs to and the outputs from the living enterprise model must be actual data from the enterprise. The model must drive the enterprise and the enterprise must drive the model. This ensures model realism and "believability".

Now that a foundation has been laid for the basis of a living enterprise model, the following dimensions of living enterprise models are proposed. Combining the like features of the requirements listed, three dimensions of a living model of the enterprise are identified (Whitman and Huff, 1997). The decompositional nature of enterprise models provides the scope dimension of the model. The model's consistency with key enterprise metrics (drives the enterprise), and the extent the model is driven from enterprise data defines the dual role of the enactment dimension. The `maintainability and the expandability of the model define the model dynamicity. The three dimensions of a living model are: scope, enactment, and the dynamicity of the model. A description of each of these characteristics follows.

Scope is the pervasiveness of the model throughout the enterprise. Enterprise modeling by its very nature is intended to provide a holistic representation of the entire enterprise. It is sometimes necessary to bound the model to a subset of the enterprise. The bounds describe the scope of the model.

Enactment is the level in which the model drives and is driven by the system. There is a wide variation in the enactment capabilities of a living model. A model can range from no enactment at all to driving the entire enterprise and providing all inputs and reporting the status of the enterprise when requested. Some more likely phases of enactment might be to use a workflow arrangement, which can provide either direction to enterprise personnel allowing them to deviate slightly from the process or require strict adherence to the process.

A model that is dynamic is able to respond to both permanent and temporary process changes to the system. As has been previously discussed, an important living characteristic of an enterprise model is its ability to change. This dimension denotes this ability. Most models today are not easy to change. The phases of dynamicity range from no capability to the model changing itself. A model could change itself by being capable of learning from its environment and then modifying itself to reflect and implement the new process (Wood, 1994). This dynamic dimension is not to be confused with simulation models, which are often called dynamic representations.

4 ENTERPRISE MODEL CLASSIFICATION METHODOLOGY

The main focus of this research is the understanding and comparison of enterprise models. However, this does not include the ranking of a model against another model. That is why the proposed metric for a living model of the enterprise is multi-dimensional rather than a single value. In other words, this is not an attempt at a single living model of the enterprise unity metric. This research is also not about defining a metric that defines the various attributes of a model regarding complexity, maintainability, or interfaceability; although these would be excellent topics for future research. Rather the primary intent of this research is to propose a methodology for classifying enterprise models to provide a mechanism for the common understanding of what the model is and thereby provide a starting point for recognizing opportunities of improvement in the model. It should be noted that the authors do not propose that all models "be all things to all enterprises." Rather, an understanding of what the model is as well as what it is not will lead to more useful models in the business of the enterprise.

The remainder of the paper discusses the proposed approach to classify enterprise models with the four dimensions: view, scope, enactment and dynamicity discussed previously. This is shown in equation 1 below. The view is designated by the first letter of the view used; activity, process, information, resource, or organization. The scope is designated as shown in Table 2 based on the three categories of processes. The dynamicity dimension refers to how often the model is updated. The scale is listed below in Table 3. The final dimension of enactment is divided into the two aspects of enactment, how much the model drives the enterprise and how much the model is driven by the enterprise. Again, the scale for this dimension is shown in Table 4 below.

$$V (S, D, E) \quad (1)$$

Where: V is the view (A, P, I, R or O), S is the scope level, D is the dynamicity level, and E is the enactment level. Tables 2, 3, and 4 provide the details of the characteristics of each level.

Table 2: Scope levels of a living model of the enterprise

Level	Characteristic
Multiple Enterprises (5)	• All three process categories are modeled across multiple enterprises
Multiple Division (4)	• All three process categories are modeled at multiple enterprise sites
Enterprise (3)	• All three process categories are modeled
System (2)	• Models are increasingly considered an asset and are therefore required for major single time decisions
Initial (1)	• Models are not considered an asset and are therefore needed only for single small decisions

Table 3: Dynamicity levels of a living model of the enterprise

Level	Characteristic
Multiple Enterprises (5)	• All three process categories are modeled across multiple enterprises
Multiple Division (4)	• All three process categories are modeled at multiple enterprise sites
Enterprise (3)	• All three process categories are modeled
System (2)	• Models are increasingly considered an asset and are therefore required for major single time decisions
Initial (1)	• Models are not considered an asset and are therefore needed only for single small decisions

Table 4: Enactment levels of a living model of the enterprise

Level	Characteristic
Optimizing (5)	• A suite of models is used which both drive and are driven by the enterprise in a systematic manner.
Managed (4)	• A formal plan is in place for models to drive *and* be driven by the enterprise when deemed appropriate.
Defined (3)	• A formal plan is in place for models to drive *or* be driven by the enterprise when deemed appropriate.
Ad Hoc (2)	• Models are driven/Models drive infrequently when convenient (less than once a year)
No Enactment (1)	• Models are not considered an asset and are therefore needed only for single small decisions. Subsequently, models do not drive and are not driven by the enterprise.

Two examples are now provided of the enterprise model classification methodology. The first is an activity model and the second is a process model. Both examples are taken from previous work at the Automation and Robotics Research Institute.

The activity model used is from a modeling effort of an aerospace company performed by ARRI in 1995. It is fairly easy to determine the scope of the model, as its A0 level activities are direct enterprise, manage assets, acquire customers/orders, design products/processes, and fill orders. This model was one of the base models for the manufacturing enterprise reference model (Whitman, 2001a) and it is easy to recognize that only the support product activity is not listed. The model is of a single division and therefore a value of 3 is assigned for its scope. This model is now four years old and has been updated less than once a year, which is when it is convenient to update the model; so, a dynamicity value of 2 is assigned. The model provides input to the enterprise on a quarterly basis and therefore R is 2. So, the enterprise model values are:

$$A (3, 2, 2) \quad (2)$$

The process model used is from a modeling effort of the same aerospace company performed by ARRI in 1998. It is fairly easy to determine the scope of the model as it is of very limited scope. The only activity addressed by this model (and that only partially) is the fill orders activity. The model is considered valuable to make a one-time decision. Therefore, this model has a scope of 2. This model is still less than one year old and has been updated once, so it is assumed that the model will be updated on an ad hoc basis so a dynamicity value of 2 is assigned. The model provides input to the enterprise on a one-time basis and therefore R is 1 for no enactment. So, the enterprise model values are:

$$P\ (2, 2, 1) \quad (3)$$

These examples demonstrate how a modeler how the resultant methodology can communicate the use of different models with different intents.

5 CONCLUSION

This paper presented a classification methodology for a living model of the enterprise. It presented the basis of a living enterprise model and the dimensions of living models of the enterprise. Scope is the pervasiveness of the model throughout the enterprise. Enterprise modeling by its very nature is intended to provide a holistic representation of the entire enterprise. The model's consistency with key enterprise metrics (drives the enterprise), and the extent the model is driven from enterprise data defines the dual role of the enactment dimension. The maintainability and the expandability of the model define the model dynamicity. This classification methodology for a living model of the enterprise can serve as a tool for enterprise engineering. The methodology will allow for comparison and evaluation of various enterprise models. A direct benefit of this research is a more clear understanding of how the enterprise modeling community uses enterprise models.

The result of this methodology is a clear and consistent understanding of the nature and use of enterprise models in a specific enterprise. By articulating taxonomy of models within an enterprise, model reuse is enabled.

6 FUTURE DIRECTIONS

Future research will apply these concepts to a collection of models of an enterprise. This paper has now established a clear distinction between Enterprise architectures and frameworks in lucid manner. The need now is to develop a general architecture for the general sector and frameworks for catering to specific enterprises. This can be done by selecting the appropriate dimensions and views. The future enterprise architectures and frameworks developed should be agile enough to cater to the constantly changing market demands.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. EPS-9874732 and matching support from the State of Kansas.

REFERENCES

EAP-Electronic Architecture Planning. `http://cio.gov/docs/committeeinfo.html`

Insurance Application Architecture -- Enterprise modeling. `http://houns54.clearlake.ibm.com/solutions/global/gfspub.nsf/detailcontacts/Insurance_Application_Architecture_IAA_`

Bernus, P., and Nemes, L. (1996). *Modeling and methodologies for enterprise integration.* Chapman and Hall.

Bernus, P., Nemes, L., and Williams, T. (1996*). Architectures for enterprise integration*. Chapman and Hall.

Curtis, B., Kellner, I. M., and Over, J. (1992). Process Modelling. *Communications of the ACM,* 35(9).

Huff, B. L., Liles, D. H., Howell, B. J., and Sanders, F. M. (1991). A Dynamic Enterprise Model for Small Manufacturers. In Proceedings of the Summer Conference on Modeling and Simulation. Pittsburg, PA.

Inmon, W., Zachman, J.A., and Geiger, J.G. (1997). *Data stores data warehousing and the Zachman framework, Managing enterprise knowledge.* McGraw-Hill.

Mills, J., and Kimura, F. (1999). *Information Infrastructure Systems for Manufacturing.* Kluwer Academic.

Nathan, B., and J. Wood. (1991). The use of IDEF0 to Document a methodology - a Novices Point of View, Automation and Robotics Research Institute, Fort Worth, Texas.

Presley, A. R. (1997). A Representation Method to Support Enterprise Engineering. Doctoral dissertation, Department of Industrial Engineering, The University of Texas at Arlington, Arlington, TX.

Presley, A., Huff, B., and Liles, D. (1993). A Comprehensive Enterprise Model for Small Manufacturers. Los Angeles, CA.

Reimann, M. D., and Sarkis, J. (1996). An integrated functional representation of concurrent engineering. *Production Planning and Control,* 7(5), 452-461.

Scheer, A. W. (1989). Business process engineering: Reference models for industrial enterprises. Berlin: Springer - Verlag.

Snodgrass, N. (1993). *Integrating Activity Based Costing with IDEF Modeling*. D. Appleton Co.

Sowa, J., and Zachman, J. (1992). Extending and formalizing the framework for information systems architecture. *IBM Systems Journal,* 31(3), 590-616.

Vernadat, F.B. (1992). *CIMOSA - A European Development for Enterprise Integration.* Massachusetts: The MIT Press.

Whitman, L. E., and Huff, B. L. (1997). A Living Enterprise Model. Miami Beach, FL.

Whitman, L. E., Liles, D. H., Huff, B. L., and Rogers, K. J. (2001a). A Manufacturing Reference Model For The Enterprise Engineer. *The Journal Of Engineering*

Valuation And Cost Analysis: Special Issue On Enterprise Engineering [To be published in 2001].

Whitman, L. E., and Huff, B.L., (2001b). On The Use Of Enterprise Models, *International Journal Of Flexible Manufacturing Systems – Special Issue on: Business Process Design, Modeling, and Analysis,* 13(2), 195-198.

Wood, J. T. (1994). Organismic Modeling of Organizations: A Dynamic Enterprise Model. Arlington: University of Texas at Arlington.

Zachman, J. (1987). A Framework for Information Systems Architecture. *IBM Systems Journal*, 26(3), 276-292.

AUTHOR BIOGRAPHIES

LARRY E. WHITMAN is an Assistant Professor of Industrial and Manufacturing Engineering at Wichita State University. He received his Ph.D. from The University of Texas at Arlington. His research interests are in enterprise engineering, supply chain management, and lean manufacturing. He is a member of IIE, APICS, ASQ, and SME. His email address is `<larry.whitman@wichita.edu>`.

KARTIK RAMACHANDRAN is a Graduate Student in the Industrial and Manufacturing Engineering at Wichita State University. He received his B.S. from Manipal Institute of Technology, India. His research interests are in enterprise engineering, supply chain management, and lean manufacturing. He is a member of IIE and APICS. His email address is `<kartikramachandran@yahoo.com>`.

VIKRAM N. KETKAR is a Graduate Student in the Industrial and Manufacturing Engineering at Wichita State University. He received his B.S. from Pune University, India. His research interests are in enterprise engineering, supply chain management, and lean manufacturing. He is a member of IIE, APICS and ASQ. His email address is `<vnket@hotmail.com>`.