The concept of enterprise integration

T.J. Williams, P. Bernus and L. Nemes

2.1 THE PURPOSE ASSIGNED TO THE TASK FORCE

Managers and information scientists, control engineers and others have long been intrigued by the concept that one should be able to correlate and use, at any one time, all of the information then available concerning an enterprise, to systematically improve the operability, profitability, agility, competitiveness (and many other important characteristics) of these systems.

Such a desire was, for example, the basis for the development of the field of computer integrated manufacturing which is built upon the ability of the computer to handle vast amounts of data or information in very short periods of time.

However, although many projects have been initiated in many types of industries in recent years to try to achieve these dreamed-of-benefits, the results achieved, in many cases, have been disappointing at best. In addition, where successes have been achieved, no clear path for obtaining future successes in subsequent applications has become clear.

The following is an important summary of the state-of-the-field of Enterprise Integration (including CIM), and points out what had been learned prior to the work of the Task Force as described herein.

2.2 THE STATUS OF ENTERPRISE INTEGRATION

An Integrated Manufacturing System solution cannot be bought off the shelf; each firm must be involved in devising its own, which explains why methodologies must be made available so that CIM systems can be built.

Designing a CIM system meets with a number of difficulties:

- The system is extremely complex, so some special techniques must be used to understand this complexity in order to efficiently act on it and to

10 *The concept of enterprise integration*

define the possible intervention area of every operator related to their skills.

- The system must take into account not only the technical points of view but also the economic, social, and human points of view in an integrated way.
- The knowledge which is necessary to design a system cannot be found in a single person; designing requires team work.
- The initial status of the system is not a matter of chance; this initial status must therefore be taken into account in order to well understand those specific constraints under which the system operates on the one hand and, on the other hand, to avoid the redesign of those parts of the system which are satisfactory. Because of today's economic situation, this point becomes more and more relevant; many industrial systems must evolve faster and faster but with as low a cost as possible.

The term 'methodology' as used here means a consistent set of components which are:

- A reference model globally and generically showing the structure of the project system to be studied.
- One or more modeling formalisms enabling the build up of the model in order to study and evaluate it.
- A structured approach for the overall program leading step by step from an existing system to a future system taking into account evolution objectives and specific constraints
- Performance evaluation criteria with which the system can be evaluated in relation to several points of view (economics, reliability, etc.).

Generally speaking, a structured approach or methodology is a set of steps to be followed to solve a problem. Within the framework of an integrated manufacturing system design methodology, the structured approach must cover all of the life-cycle of the integration project which is split into several successive stages (analysis, design, development, implementation, and operation). Every step of the methodology must be precisely defined and based on a standardized project structure giving the set of actors for which the work must be precisely defined as well.

Defining very precisely all steps which are required to go from concept to specification to implementation is essential. During each step some models are built and checked. The consistency of these models and results for each step must also be constantly checked [1].

As used in the context above, an architecture is any method (drawing, model, description, etc.) for giving the structure or framework showing the interrelationship of all of the parts and/or functions of a device, system or

enterprise. A reference architecture is a collection of the overall generic functions, descriptions, or behaviors of many (hopefully all) types of systems and their associated structures or frameworks.

Thus an architecture, or preferably a reference architecture, should form the basis for the development of a device, system or project for carrying out an information integration program for a manufacturing or other enterprise. The problem remains, however, of: What is the best architecture for such purposes?, and How may it be developed? This then was the task assigned to the Task Force which developed the report contained herein.

The above statement characterizes very succinctly the industrial requirements which needed to be considered by the IFAC/IFIP Task Force on Architectures for Integrating Manufacturing Activities and Enterprises as it faced the task assigned to it by its parent bodies. Thus our task of evaluating manufacturing integration architectures must consider not only the 'what' of integration but also the 'how' of integration as well. That is, how integration is accomplished in terms of the development of the system, not just the overall design of the computer system (for example) needed to achieve integration.

This line of reasoning is reinforced when one reviews the field of integration architectures. An architecture, as just noted, can be defined as a description (model) of the structure of a physical or conceptual object or entity. Thus there are two and only two types of architectures which deal with the integration of manufacturing entities or enterprises. These are:

- The structural arrangement (design) of a physical system such as the computer control system part of an overall enterprise integration system.
- The structural arrangement (organization) of the development and implementation of a project or program such as a manufacturing or enterprise integration or other enterprise development program.

All of the reference architectures discussed in this book should therefore be of the second type in order to fulfill the requirements voiced above. (It is noted in passing that architectures of the first type can well become major subunits of architectures of the second type, for instance as examples or guides in the functional or detailed design phase of the integration program development).

There were only three major architectures in the open literature which were known to the Task Force as of this writing which fall into Category 2 above. These were:

- The Open System Architecture for Computer Integrated Manufacturing – CIMOSA, as developed by the European CIM Architecture Consortium (AMICE – a backward acronym) under ESPRIT Projects 688, 2422 and

5288 of the European Community. This work was initiated in 1984 (hereafter called CIMOSA). [3]

- GRAI-GIM. The GRAI Integrated Methodology as developed by the GRAI Laboratory of the University of Bordeaux in France. This work resulted from production management studies initiated at the GRAI Laboratory as early as 1974. It has taken its current form since about 1984 (hereafter called GRAI-GIM). [1,2]
- The Purdue Enterprise Reference Architecture and the related Purdue Methodology as developed at Purdue University as part of the work of the Industry – Purdue University Consortium for CIM. This latter work started formally in 1989 but bears on the Purdue Reference Model developed in 1986 and earlier work of the Purdue Laboratory for Applied Industrial Control dating back to the mid seventies (hereafter called Purdue or PERA). [4,5]

These then are the three architectures which were extensively examined by the Task Force and which are thoroughly described and evaluated in this book.

It must also be remembered that manufacturing integration is not automation and/or computer application per se, although computers and automation are usually involved in its implementation. Manufacturing or enterprise integration, on the one hand, is the collection, reduction, storage and use of data (and/or information) from the business entity involved, plus its environment, in order to optimize the operation of the business entity as a whole according to the criteria established by that business entity's management. On the other hand, it must also include the integration of the raw material, intermediate and final product flows, as well as machine organization, to further the enterprise optimization possible. In addition, the actions and influence of the human staff involved must also be thoroughly integrated with the manufacturing or other enterprise operations as well as the information integration carried out by the computers and related devices.

2.3 THE RELATIONSHIP OF THE TASK FORCE'S WORK TO THAT OF OTHERS

It is the purpose of this book to define enterprise reference architectures; present what they are and what they can be used for; treat their various forms; and project the future of the field of enterprise integration as it appears in the findings and recommendations developed by the Task Force in its investigation. The Task Force believes that it has developed new and

important concepts in its work. These are presented herein. The Task Force further believes that these concepts, findings and conclusions may be helpful to others as follows:

Information technology experts, who develop and market comprehensive business information systems (software or hardware), information systems development methods, and software engineering or CASE tools, may find guidance on how to complete presently available but unfinished enterprise system architectures so that these architectures can become truly generic, and therefore applicable to the entire enterprise, and indeed to all types of enterprises regardless of the industry or type of endeavor involved.

The state of the art of enterprise reference architectures is presented here in order to allow the owners of proprietary architectures to assess their own products in view of this present description of the generic architecture requirements.

Developers of control systems (both hardware and software) will hopefully find that this book can also give them guidance in a way similar to that for the information technology experts. The control engineers situation differs in our view only in that they cover a different subpart of the entire enterprise domain from the information technologist. That is, they concentrate particularly on control system description techniques, methods for control system design, and/or tools to support these various methods rather than on data handling techniques, data base design, etc.

Developers and users of life-cycle methodologies in consulting firms, standards bodies, or large multinational companies may find our work useful to them to help them to evaluate their currently applied methodologies in the light of the analysis given herein. This should then help them to formulate their own future strategies to allow courses of action which would lead to a more complete enterprise integration methodology than those available today.

Policy makers and funding agencies may also find here a set of overviews and evaluations which they can then use for assessing ongoing research results. Our results may also be useful for deciding on the future research directions they may wish to pursue.

Figure 2.1 Representation of the separate classifications of architectures as used in this book.

2.4 A FURTHER DEFINITION OF THE RELATIONSHIP BETWEEN ARCHITECTURES

Fig. 2.1 presents a diagram which illustrates the way in which the Task Force has defined the relative scope of Type 1 and Type 2 Architectures. It will be seen that the Type 2, Enterprise Reference Architecture, comprises the whole context of the diagram of Fig. 2.1. That is, it presents a life-cycle development methodology for all aspects of the enterprise concerned.

The Type 1, CIM Architecture, on the other hand comprises only those aspects involved in the computer-based information technology of the enterprise, i.e. as noted here, those aspects included under the categories of Business Processing and Engineering and Control.

2.5 MAIN FEATURES OF ARCHITECTURES

As shown by Fig. 2.1 we have subdivided all enterprise activity into the following three areas.

- The Production Facility
- Human Organization Including
Management
Staff, and
Production Functions
- Information Technology (Hardware and Software) including
Business Information
Engineering Systems
Production Management, and
Control Systems

What makes this subdivision useful is that it not only reflects distinct boundaries between the different types of enterprise activity but it also points out disciplinary (or cultural) boundaries. Using this division we can easily identify research and development results, vendors' product families, or standards which may be targeting to one or the other of the above areas.

For example, standards and product families have been developed for the integration of business information systems separately from the rest of the enterprise and are offered as solutions there because they have a generic structure which is applicable to virtually any case which could be considered. Similarly, large control system vendors have standards and product families which cover process control, product design, and production management, etc. The pattern is the same: these standards and families of products are based on some generic structure which is applicable to virtually any case of industrial control and engineering.

Although many of these generic structures are of use only in such well defined subdomains of all enterprise activities as those just noted they are nevertheless essential points of departure for the integration of the entire enterprise.

However, in order to realize the full potential of the use of each of these separate solutions one has to be able to form a unified structure that enables the integration of the information flow between these domains just as easily as within the confines of the domains themselves.

Type 1 Architectures (CIM Architectures) are for example a partial solution to such generic structures (see Fig. 2.1) since they allow the integration of the respective information technology regimes, such as business information processing system products with engineering (CAD, CAM or Con-

trol System) products thus describing a complete structure for the computer control system of the enterprise.

Given that either of these domains (business processing or engineering control (see Fig. 2.1) can separately define their own respective information processing, database, communications and human interface solutions (thus structures and standards), a CIM architecture which encompasses the control of all enterprise activities should have a common base rather than two separate solutions for the above mentioned subdomains. Type 1 architectures (or CIM architectures) as structural descriptions of the complete enterprise control system must therefore include traditional business processing as well as engineering, control and management in order to provide a proper solution to this requirement.

Most CIM Architectures known from the literature are of this Type 1 Classification and an overview of these is given in Chapter 5 of this book (e.g. [4,6] and many others, see references made there).

It is expected that the increasing drive to comply with open systems standards¹ and the adoption of client-server structures, will soon make it possible to establish enterprise-wide communication and processing infrastructures or Integration Platforms which will serve the needs of both the business processing and the engineering (engineering control and management) functions. Such integration platforms have already been developed for many of these limited domains. The remaining work is to extend them to complete the required overall generic platforms.

For example, IBM's System Application Architecture (SAA) [7,8,9] is such an intermediate platform devoted specifically to the domain of business processing. Any such integration platform, as a collection of hardware and software solutions, is preferably based on 'open systems' standards, allowing configuration flexibility and multivendor, vendor independent configuration of systems. These platforms must include a standard operation system interface [10], communication protocols [11,12], a user interface [13,14], a database access language [15], remote database access [16, 17], and distributed computing (client server computing) [18,19].

As an example of the overall capability desired, the Integrating Infrastructure of CIMOSA [3], which is under development, is intended to be a platform for the entire domain. Where practicable the AMICE Consortium developing CIMOSA has made a commitment to adopt existing standards such as those discussed above to promote these concepts.

1. Open systems standards include communications, operating systems, database access, distributed processing, and user interfaces.

2.6 TYPE 2 ARCHITECTURES IN RELATION TO TYPE 1

However, as is made abundantly clear by Fig. 2.1, the above discussion only comprises Item 3 of the list of three given in this section. A complete enterprise description by our definition must also comprise the integration of all human relations and human organizational activities along with those of the manufacturing facility, plus those of the computerized CIM system to have a completely integrated enterprise. Methods for carrying out the design work for each of the three domains are represented in Fig. 2.1 as horizontal patches adjoining the respective structural descriptions. Examples which may be given are: in the domain of business processing – A/D Cycle [20]; in the domain of production facility design – methods based on group technology [21,22,24] and on layout design techniques [23, 25]; in the domain of generic software engineering development – see [9, 26, 27]. There are also many different methods available for (separately) developing the human organization and management aspects of an enterprise.

An important message which will be derived from a reading of this book is that the simultaneous coverage of all of these areas is not yet available and that there are as yet only a very few (although quite important) attempts known in the open literature to try to develop such coverage. However, many groups worldwide are hard at work to correct this situation and important results can be expected soon.

We believe that the possibility to build comprehensive design methods in each area will accentuate the need for a complete life-cycle architecture to coordinate their comprehensive use together in a complete methodology. Naturally, such a development cannot be independent from the methods already developed for the separate domains as presented above. The substantial investment in these areas makes it necessary for the complete enterprise reference architectures to include, adopt and adapt these existing methods and tools as substantial building blocks in forming the overall complete methodology.

An overall and completely generic Enterprise Reference Architecture is thus an umbrella under which each of these separate disciplines and their methods can be integrated together with the others needed and one of the contributions which this book can make is to point out how such an architecture can be developed.

2.7 ORGANIZATION OF THIS BOOK

Chapters 3–5 present some background and supporting information concerning the integration architectures and modelling field necessary for our study. Chapter 3 defines the needs and requirements for architectures for integrating manufacturing activities and enterprises and thus establishes the goal against which our evaluations will be carried out. Chapter 4 presents a history of the field. Chapter 5 gives an extensive bibliography and discussion of the available architectures and models.

Chapters 6–8 describes the candidate architectures to be evaluated: CIMOSA (Chapter 6), GRAI-GIM (Chapter 7) and Purdue (Chapter 8).

Chapters 9–18 present the evaluation methods used and the direct results obtained. Chapter 9 introduces the background of the methods employed. These methods were: a detailed Questionnaire (Chapters 10–13), its companion Short Form (Chapter 15), the answers to these as evaluated in Chapter 14; the Direct Mapping – One on One (Chapter 16); and the Matrix of Requirements Mapping (Chapter 17). An evaluation of the results obtained is presented as part of each chapter. Chapter 18 presents some additional observations concerning each of the architectures. Chapters 19 and 20 contain our Conclusions and Recommendations. First there is a compendium of the Conclusions derived from each of the evaluation methods and our analysis of these. Next we present our considered recommendations for the future of these architectures and for the development of the field itself. Also presented is a proposal for future work needed to further the field of enterprise integration and of architectures describing the field.

Appendix A contains the history of the Task Force and its work program, Appendix B lists its members in the first triennium of the Task Force's existence.

2.8 REFERENCES

- [1] Doumeingts, G., Vallespir, B., Darracar, D. and Roboam, M., 'Design Methodology for Advanced Manufacturing Systems,' *Computers in Industry*, Vol. 9, No. 4, pp 271-296 (1987)
- [2] Doumeingts, G., Vallespir, B., Zanittin, M., and Chen, D., GIM-GRAI Integrated Methodology, a Methodology for Designing CIM Systems, Version 1.0, LAP/GRAI, University Bordeaux 1, Bordeaux, France (May 1992)
- [3] AMICE Consortium, CIMOSA Open Systems Architecture for CIM, Springer Verlag, Berlin (1993)

- [4] Williams, T. J. (Editor), A Reference Model for Computer Integrated Manufacturing (CIM), ISA Publications, Research Triangle Park, NC (1989)
- [5] Williams, T.J., The Purdue Enterprise Reference Architecture, ISA Publications, Research Triangle Park, NC (1992)
- [6] Thacker, R. M., A New CIM Model – A Blueprint for the Computer-Integrated Manufacturing Enterprise, Society of Manufacturing Engineers, Dearborn, MI (1991)
- [7] Anonymous, Special Issue on IBM's Systems Application Architecture (SAA) IBM Systems Journal, Vol. 27, No. 3 (1988)
- [8] Radesi, S. J. and Czubek, D. H., SAA, IBM's Systems Application Architecture, Van Nostrand Reinhold, New York (1991)
- [9] Thacker, R. M. and Dorfmann, M., 'System and Software Requirements Engineering', IEEE Computer Society Press Tutorial, p719 (1990)
- [10] Lewine, D. A., Posix Programming Guide: Writing Portable Unix Programs With the Posix 1 Standard, O'Reilly & Assoc., Sebastopol, CA, pp 99 (1991)
- [11] Anonymous, ISO International Standard DIS 7498 Data Processing – Open Systems Interconnection – Basic Reference Model, International Standards Organization (December 3, 1980)
- [12] Anonymous, Manufacturing Automation Protocol, Version 3.0, General Motors Corporation, Society of Manufacturing Engineers, Detroit, MI (1987)
- [13] Berlage, T., OSF/Motif: Concepts and Programming, Addison-Wesley, Reading, MA (1991)
- [14] Davidson, A., Distributed Window Systems, A Practical Guide to X11 and Open Windows, Addison Wesley (1992)
- [15] Anonymous, American National Standard for Information Systems – Database Language – SQL, ANSI Publication # X3.135-1986
- [16] Anonymous, ISO Draft International Standard, Information Technology – Open Systems Interconnection – Remote Database Access, prepared by the ISO/IEC JTC1/SC21/WG3 (1992)
- [17] Arnold, D., et al. 'SQL ACCESS: An Implementation of the ISO Remote Database Access Standard,' Computer (1991)
- [18] Anonymous, Distributed Computing Environment (DCE), Open Software Foundation, Inc. (OSF) (1992)

- [19] Anonymous, ISO International Standard 10746 Basic Reference Model of Open Distributed Processing (ODP)¹ Also as ITU-T (CCITT) Draft Recommendations for Open Distributed Processing X.900 (X.901, X.902, X.903 and X.904) In preparation: (July 1993)
- [20] Anonymous, Special Issue on A/D Cycle, IBM Systems Journal, Vol 29, No. 2 (1990)
- [21] Gallagher, C. C., Group Technology Production Methodology in Manufacture, Ellis Horwood, Chichester (1986)
- [22] Ham, Inyong, Hitomi, Katsuo and Yoshina, Teruhiko, Group Technology: Application to Production Management, Kluwer-Nijhoff, Dodrecht (1985)
- [23] Domschke, W. and Drexl, A., Location and Layout Planning, in International Bibliography, Springer Verlag, Berlin (1985)
- [24] Snead, Ch. S., Group Technology: Foundation for Competitive Manufacturing, Van Nostrand Reinhold, New York (1989)
- [25] Hales, H.L., Computer-aided Facility Planning, Marcel Dekker, New York (1984)
- [26] Humphreys, W. S., 'The IBM Large Systems Software Development Process: Objectives and Directions', IBM Systems Journal, Vol. 24, No. 2, pp 76-78 (1985)
- [27] Radice, R. A., Roth, N. K., O'Hara, A. L., Jr. and Ciarfella, W. A., 'A Programming Process Architecture', IBM Systems Journal, Vol. 24, No. 2, pp 79-90 (1985)

2.9 FURTHER READING

- Anonymous, ISO International Standard Information Technology Software Life-Cycle Process, In preparation: IEC/ISO (JTC1) - SC7²
- Date, C. J., A Guide to DB2, Addison Wesley, Reading, MA (1988)
- Hoffnagel, G. F. and Beregi, W. 'Automating the Software Development Process', IBM Systems Journal, Vol. 24, No. 2, pp 102-120 (1985)
- Martin, J. K., Chapman, Kavanaugh and Legen, J., SAA: Common Communications Support, Network Infrastructure Prentice Hall, Englewood Cliffs, NJ (1992)

1. As of August 1993 the ODP standard is in preparation. This four-part standard will consist of: Basic Reference Model of Open Distributed Processing: Part 1 Overview and Guide to Use, – Part 2: Descriptive Model, – Part 3: Prescriptive Model, – Part 4: Architectural Semantics.

2. As of August 1993