

Preface to Third Edition

The most critical issue facing government, defense, and commercial enterprises today is the rapid pace of change in almost every industry. With the rate of technological change increasing, together with today's budget and competitive pressures, enterprises must be able to change rapidly ... often just to survive—let alone to succeed.

The need to transform from today's inflexible business environment to an agile enterprise that can change direction rapidly has never been greater. Yet the structures, processes, and systems that we have today are inflexible: they are incapable of rapid change. And more computer hardware, or software, or packages, or staff, or outsourcing is not the solution. They are part of the problem.

The solution requires methods and technologies for rapid business change—with systems that also change in lock step. This is *not* a computer problem. It is a business problem, one that needs strategic direction from senior management and strategic planners, with these directions then translated into rapid action by business experts working with IT experts.

What are needed are methods that enable senior managers—together with their planners, business managers, business experts, and IT staff—to work together to achieve business change, with each group contributing its specific expertise. The methods to achieve this are being successfully applied by many enterprises today. But these methods need new thinking. The tried and true ways are not fast enough. We need new ways to make the required business change transformations.

Our current systems development methods have served us well for developing operational information systems in the period of managed change that we had up until the 1990s. But now the pace of change is much faster than we ever anticipated when those systems were first built.

Historically, these systems have been difficult to change. The systems and databases that we built in the early years of the Information Age to enable our organizations to be more responsive to change are now monolithic and resistant to change. Today, they inhibit the ability of our organizations to change rapidly in order to compete ... sometimes even to survive. We are chained to inflexible systems that no longer respond to the rapid change environment of today—let alone the even greater change environment that we will find ourselves in tomorrow.

We need to build more flexible systems for the future that can change easily, rapidly, and often. To achieve this, the systems development methods that we use should take a different focus for the future. They must be able to

identify potential future changes early. We must also build systems and databases differently, so that they can be changed rapidly to support vital business changes. These changes must be capable of being made within weeks, even days—not in years, as is the case today. This book addresses enterprise integration using enterprise architecture methods and technologies. Enterprise architecture achieves *business integration*. It requires a focus on the future: through methods for strategic business planning, for creating balanced scorecards, and for governance. These strategic planning methods are covered in Part I.

Business integration is also achieved by enterprise architecture methods that address the integration of data, processes, locations, people, events, and motivation for an enterprise. Enterprise Architecture (EA) methods are briefly introduced in Chapter 1; they are covered in detail in Part II of the book, with methods to identify priority systems and databases for rapid delivery into production in 3-month increments.

Enterprise integration also includes *technology integration*—using the technologies of extensible markup language (XML), enterprise application integration (EAI), enterprise portals, Web services, and service-oriented architecture (SOA) with business process management (BPM) languages such as Business Process Execution Language (BPEL) that are automatically generated from process or workflow diagrams. These technologies can be used to deliver priority systems and databases rapidly into production in 3-month increments and are covered in Part III of the book.

To be able to succeed, we also need *methodology integration*. The methods covered in Part II have been defined so they support each other, integrating data and accessing it from reusable processes that are *automatically* identified from data models—along with project plans so these reusable processes can be implemented and delivered into production rapidly.

We are at a dramatic and historical point of convergence: in business and in technology. The Internet and associated technologies today enable all of the customers, suppliers, and business partners of an enterprise to work together at electronic speeds. These technologies are transforming organizations. Processes that took days or weeks to complete previously by using mail, fax, and courier communications now take hours, minutes, and sometimes – even seconds. This is the direct consequence of technology.

But technology alone is not the answer. To achieve any degree of success in enterprise integration, technology integration must be used within a coherent, integrated enterprise, through business integration. Most enterprises still have a long way to go to realize business integration.

To appreciate what still has to be achieved, we need to review what I call the *process-engineering bible*. I describe it in this way because it has had a dramatic effect on the way in which organizations function. To consider its impact, we need to review its message. But first:

- What is its title?

- Who was the author?
- When was it published?

Perhaps we can identify the book by first considering its author:

- Was it Michael Hammer or James Champy of *Reengineering the Corporation* [¹] fame? No, it was neither of them.
- Was it Ken Orr [²], Ed Yourdon [³], or Tom de Marco [⁴] of *Software Engineering* fame? No, it was not them either.
- Well, was it Peter Drucker of Management [⁵] and Strategic Planning [⁶] fame? No, not him.
- Was it W. Edwards Deming of quality control fame? No, not him either.
- Was it Alfred Sloan or Henry Ford? No, the book I am referring to was published long before all of these eminent people.

So which book am I talking about? As soon as I give you the author and its title—with its publication date—its significance will become apparent. The reference is as follows:

- Adam Smith, *Wealth of Nations* (1776) [⁷]

This was one of the most influential books at the start of the Industrial Age. It described the evolution from the Agricultural Age to the Industrial Age. It was the foundation for most industrial enterprises in the late 18th century and into the 19th century.

To understand the importance of Smith's *Wealth of Nations*, we will review part of his first chapter. Box P.1 provides an extract from Chapter 1 of *Book One*. Its language is unusual today. I have included part of the initial paragraphs; to help readability I have added comments in parentheses to indicate the terminology that we use today to describe the same concepts.

The principles that Adam Smith advocated broke complex processes into simpler process steps. He showed, by using technologies available in his day, that an illiterate workforce could be trained to carry out each step repetitively. In this way they were able to achieve much higher levels of productivity than if one worker carried out each step in turn. Smith showed that component steps could also be combined in different ways for new, improved processes. These are the

¹ Hammer, M., and J. Champy, *Reengineering the Corporation*, London: Nicholas Brealey Publishing, 1993.
² Orr, K., *Structured Systems Development*, New York: Yourdon Press, 1977.
³ Yourdon, E., and L. Constantine, *Structured Design: Fundamentals of a Discipline of Computer Program Systems Design*, Englewood Cliffs, NJ: Prentice-Hall, 1978.
⁴ De Marco, T., *Software Systems Development*, New York: Yourdon Press, 1982.
⁵ Drucker, P., *Management: Tasks, Responsibilities, Practices*, New York: Harper & Row, 1974.
⁶ Drucker, P., *Management Challenges for the 21st Century*, New York: HarperCollins, 1999.
⁷ Smith, A., *An Inquiry into the Nature and Causes of the Wealth of Nations*, 1776. Often just called "Wealth of Nations".

same concepts that we still use today for *reusability*, using *object-oriented methods*.

Box P.1: Extract from Adam Smith’s “Wealth of Nations”⁷

EXTRACT FROM BOOK ONE: “*Of the Causes of Improvement in the Productive Powers of Labour, and of the Order According to which its Produce is Naturally Distributed Among the Different Ranks of the People.*”

CHAPTER 1: “*Of the Division of Labour*”

“... To take an example, therefore, from a very trifling manufacture; but one in which the division of labour has been very often taken notice of, the trade of the pin-maker ... a workman ... could scarce ... make one pin in a day, and certainly could not make twenty. [In today’s terminology he is referring to *serial operation*.]

But in the way in which this business is now carried on, not only the whole work is a peculiar trade, but it is divided into a number of branches of which the greater part are likewise peculiar trades. [In today’s terminology this refers to *object-oriented methods*.]

One man draws out the wire, another straightens it, a third cuts it, a fourth points it, a fifth grinds it at the top for receiving the head; to make the head requires two or three distinct operations [*object-oriented encapsulation*]; to put it on is a peculiar business, to whiten the pins is another; it is even a trade by itself to put them into the paper; and the important business of making a pin is ... divided into about eighteen distinct operations [*object-oriented methods*]

I have seen a small manufactory of this kind where ten men only were employed ... they could, when they exerted themselves, make among them about twelve pounds of pins in a day ... upwards of forty-eight thousand pins in a day. Each person, therefore ... might be considered as making four thousand eight hundred pins in a day. [*object-oriented reusability*]

But if they had all wrought separately and independently ... they certainly could not each of them have made twenty, perhaps not one pin in a day [serial operation] ... ; that is, certainly, not ... the four thousand eight hundredth part of what they are at present capable of performing, in consequence of a proper division and combination of their different operations.” [*object-oriented reusability*]

Evolution from the Industrial Age to the Information Age

Adam Smith’s breakthrough was the foundation for late eighteenth-century–early nineteenth-century industrial enterprises. With their focus on manufacturing

physical items, this period also saw the same concepts applied to knowledge-based processes for bank loans and for insurance policy applications. Instead of manufacturing steps, a loan application or a policy application approval process was broken down into discrete steps to be carried out by different people, each skilled in completing an aspect of the relevant application. Each process step was carried out in a defined sequence: One step was completed before the next step in the sequence was started. The result was the definition of *serial processes*.

As the application form was routed to each person in the approval process, details of the relevant applicant and the current status of the process were recorded in handwritten ledgers; these were called the applicant ledger or the customer ledger. Each person involved in carrying out a process step kept an individual record of every applicant or customer that worker had processed, and the stage the applicant had reached in the approval process.

The twentieth century saw an improvement in these process steps with the introduction by Henry Ford of the *assembly line* method of automobile manufacture. The vehicle being built physically moved along each section of the assembly line, where different components were added in each step of the assembly process.

This period also saw the introduction of *parallel processes*, in which two or more processes could be carried out concurrently, with each process step executed independently of other process steps. An example is the parallel construction of the body of the automobile, while its engine is constructed at the same time. Each parallel process path is thus independent of the other parallel paths, until they need to converge. Only when the automobile has to be driven off the assembly line does the engine have to be fitted into the car.

By the middle of the twentieth century, the industrial enterprise had evolved into a complex series of manual processes. The pace of progress had seen most enterprises evolve to use increasingly complex business processes, with rapidly growing transaction volumes to be manually processed. And what was the result? These enterprises found *they were operating in a continual state of manual chaos!*

Then the computer came on the scene in the second half of the twentieth century. From the late 1950s—through the 1960s, 1970s, and up to today—we have seen manual processes being automated by computer. What was the result? The processes were automated, but we took the existing manual processes and then automated them essentially *as-is*, without much change. That is, the automated processes were being executed as the manual processes used to be, but faster and more accurately. *In so doing, we moved from manual chaos ... to automated chaos!*

Enterprises tried to hide this automated chaos. Through to the mid-1990s, most enterprises could confine their automated chaos to the back office. They presented a calm, in-control, front-office appearance to the outside world. They tried to emulate the graceful swan, gliding silently across the glass-like surface of a

lake with no apparent effort. The furious paddling activity—trying to move ahead—was hidden beneath the surface.

But with rapid acceptance of the Internet in the second half of the 1990s, the chaos moved from the back office onto the front doorstep of enterprises: through their web sites. *Customers could visit these enterprises with the click of a mouse. But they could just as quickly leave with the click of a mouse if they did not find what they needed!*

The reason they left is not because of what the automated processes could do; rather, they left because the processes did not provide what the customers needed. This was often due to redundant processes and redundant data, which, by definition, are non-integrated. Another term for non-integration is *disintegration*. That is, by using automation most enterprises had evolved from non-integrated manual processes to *disintegrated automated processes*.

The problem, however, is much worse than this! Most automated processes today assume that the technologies of the past still apply. The manual processes that they automated required paper-based forms that were mailed, or later faxed. So their automated counterparts are based on forms that are also printed to be mailed or faxed. On receipt at their destination, the data in these forms are manually reentered into relevant systems—with manual work, with extra staffing to do that reentry, with delays, with errors, and with associated costs.

In Part III of this book, we will see how technologies can be used to convert printed forms into electronic forms using the extensible markup language (XML). These XML forms can be transmitted electronically to receiving applications within an enterprise or between enterprises. This is called *enterprise application integration*. It replaces mail transmission and manual reentry for paper-based systems that were designed for completion over weeks or days. Instead, electronic forms and systems that inter-communicate within minutes or seconds—anywhere in the world—replace paper forms.

The problem is that automated systems that assume intercommunication with printed forms and manual reentry over weeks and days do not work well when asked to intercommunicate with electronic forms that bypass the need for manual reentry—and that are completed in minutes or seconds. What is the basic reason for this dichotomy?

Today we have twenty-first-century enterprises that utilize twenty-first-century technologies, *yet astonishingly, most enterprises today still use eighteenth-century disintegrated business processes!*

The business processes—originally designed based on principles set by Adam Smith in 1776—have not evolved to take advantage of the technologies we have available today. This is a business problem, not a technology problem. It requires business decisions. It requires business expertise. These are the basic ingredients for business integration.

Part II shows how business integration for business transformation is realized by enterprise architecture. But the real architects of an enterprise are **not** found in its IT department. This leads us to two important principles – highlighted below:

1. *Enterprise architects are the senior managers who set the directions for the future, based on processes designed for that future and its technologies. But the future cannot continue to be based on eighteenth-century business processes that no longer respond to the rapid-change environment of today ... and even greater change tomorrow.*
2. *The future will be based on business transformation through processes that use the technologies of today and tomorrow to complete in minutes and seconds what before took days and weeks ... with strategic directions set by senior management, and with business experts and IT experts working together in a design partnership.*

Enterprise architecture methods of enterprise engineering enable business experts and information technology (IT) experts to work together to identify reusable business activities, reusable business processes, and integrated databases for business integration. These take advantage of the latest technologies for technology integration—with integrated twenty-first-century enterprises that have been transformed through the use of reusable twenty-first-century processes.

Third Edition Changes

As the Father and Architect of Information Engineering (IE) and after more than 50 years in the IT industry, I have now retired in Perth, Western Australia. In the Third Edition of this book I wanted to add an Appendix, based on a paper that I wrote in 2014: “*What is Wrong with Information Engineering and How to Fix it?*”. I wanted to set the record straight on Information Engineering. There have been spectacular successes with IE, and also some dismal failures. I wanted to clear up misconceptions of how to conduct IE projects; to outline how to maximize the chances of success; and also to highlight the misconceptions and discuss the reasons why data-driven IE projects may have been disproportionately small in number, but have been spectacularly large in achievement. In contrast, only minor improvements have been achieved with process-driven IE over what have been realized using traditional Software Engineering methods.

The third edition *Appendix 1* starts by discussing many well-known problems of systems development using Software Engineering and some solutions to these problems. I describe my early experience with these problems while working for IBM over 15 years in Australia and the USA. I left IBM in 1976 to form my own company and develop an integrated methodology to address these problems. I describe the evolution of this methodology from 1976 – 1980 as a rigorous discipline – like an engineering discipline – for information; this dictated the choice of the name: Information Engineering.

The initial publications on IE were: first, an InDepth series of articles, published by Computerworld USA in May-June 1981; and second, the co-authored book that I wrote with James Martin, published by Savant Institute in Nov 1981. From this point, IE diverged as two distinct variants. Popularized by Martin, the world rapidly adopted IE throughout the 1980s. I compare these two variants: highlighting danger points to avoid; I also outline the steps to take to ensure success. In the 1980s, 1990s and into the 21st century, Information Engineering evolved dramatically from 1995 into an enhanced version called “*Enterprise Engineering*”. It has been largely automated and is now being successfully used to deliver Enterprise Architecture (EA) projects into production as integrated databases and reusable processes and systems – in 3-month increments – for small and medium organizations; and also for the largest commercial, government and Defense organizations.

Acknowledgements

I have acknowledged key individuals, by name, within Appendix 1. I would like to gratefully acknowledge the contributions made by all past and present staff of the following companies:

- Information Engineering Services Pty Ltd (IES) in Sydney and in Perth. Western Australia.
- Information Engineering Australia Pty Ltd (IEA) and later, Information Engineering Systems Ltd (IESL) in Sydney, Australia.
- Information Engineering New Zealand Ltd (IENZ) in Auckland, NZ.
- Information Methods Corporation (IMC) in Boston, USA.
- Information Engineering Systems Corporation (IESC) in Washington, DC.

Reading Strategies

This book has been designed so that each chapter stands alone and covers all of the concepts of each relevant method or technology. It has been written as a how-to text that serves the needs of a diverse audience:

- Business executives and staff (CEOs, COOs, CFOs), business managers and business experts,
- IT executives and staff (CIOs, CTOs, IT managers), project managers, business analysts and systems analysts, technical IT staff, and enterprise architects.

I will address each of these roles and interests, highlighting the parts of the book that will be of greatest interest to each group. *Changes in the second edition are discussed in separate paragraphs, in italics.*

For All Readers

The book starts with an overview of enterprise architecture and enterprise engineering, which are required reading for all readers.

Chapter 1: Evolution to the 21st Century Enterprise. This chapter provides a non-technical introduction to the basic principles of Enterprise Architecture and Enterprise Engineering from a management and an IT perspective. It establishes the fundamental principles on which the book is based. All readers should read it. The first edition of the book covered V1.0 of the Zachman Framework for Enterprise Architecture, which evolved from 1987.

The second and third editions of this chapter also cover V3.0 of the Zachman Framework with its clearer business focus. This was released in August 2011. The second edition required extensive text changes and clarification throughout Part I on Enterprise Architecture for Managers and throughout Part II on Enterprise Architecture Methods.

Business and IT Executives and Methodology Readers

Part I: Enterprise Architecture for Managers is for business managers and business experts, as well as the IT staffs who will work with them on enterprise architecture projects. The chapters in this part introduce balanced scorecard and strategy maps, strategy analysis, and governance analysis. A brief overview of each chapter is provided here and at the start of Part I:

Chapter 2: Balanced Scorecard and Strategy Maps. This chapter discusses the concepts of balanced scorecard and strategy maps, which can be used as a catalyst for business transformation. These are catalysts for Part II; using enterprise architecture methods to ensure that systems and databases provide required balanced scorecard support.

Chapter 3: Using Strategy Analysis to Define the Future. This chapter describes the strategy analysis management methodology: a rapid-delivery method for strategic business planning and balanced scorecard used by senior managers and business experts to identify requirements and to set future business directions. It is introduced with many examples, together with a business planning questionnaire template (in Word) that is provided online, for use with the book.

Chapter 4: Governance Analysis using Enterprise Architecture. Many countries have enacted legislation for corporate governance. For example, the United States' Sarbanes-Oxley Act of 2002 requires internal control reporting for senior management to ensure that financial reporting and other governance controls are in place. Enterprise architecture enables governance analysis frameworks to be dynamically defined for each enterprise for internal control reporting purposes.

Part II: Enterprise Architecture Methods covers several business-driven methods used for enterprise architecture. Each chapter fully describes relevant methodology concepts, with many case study problem exercises to test your understanding; together with sample solutions. These chapters are discussed briefly next.

The second edition of Part II included extensive text changes in each chapter to incorporate the terminology used by V3.0 of the Zachman Framework.

Chapter 5: Methods for Building Enterprise Architecture. As a Methodology overview chapter, this discusses the use of Enterprise Architecture in the USA Federal Government and the USA Department of Defense (DoD). It covers the Federal Enterprise Architecture Framework (FEAF) and the DoD Architecture Frameworks (C4ISR and DoDAF). It describes dramatic cost savings that have been achieved in past multi-year projects using enterprise architecture. The rapid-delivery EA methods enable these savings to be achieved in 3-month increments, delivering key business processes, databases and systems rapidly into production. The steps that achieve this rapid delivery are covered in detail in the remaining methodology chapters.

The second edition of this chapter incorporated extensive text changes for V3.0 of the Zachman Framework. The introduction and discussion of The Open Group Architecture Framework (TOGAF) in the first edition was expanded in the second edition.

Chapter 6: Using Business-Driven Data Mapping for Integrated Data. IT data administrators have previously interviewed business experts using data modeling. This chapter describes a business-driven data mapping method that is used by business experts and IT experts working together in a design partnership. It establishes the foundation that is essential for data integration, so that common data can be shared throughout an enterprise. The chapter uses many business examples, with case study exercise problems and sample solutions.

The second and third editions include a discussion of how this chapter can be used to study for qualification as a Certified Business Data Modeler (CBDM). It adds to the description of the Data Modeling Case Study Workshop discussed in the first edition: this Workshop is the exam to qualify as a CBDM.

Chapter 7: Strategic Modeling for Rapid Delivery of Enterprise Architecture. This is a key foundation methodology chapter. It describes Entity Dependency Analysis, which is used to identify reusable business activities and business processes from data maps. It enables project plans to be derived manually and also automatically from data maps, documented as activity clusters. This method enables high priority business sub-projects to be identified for delivery in 3-month increments. This method has been used over the last 25 years as an integral part of business-driven enterprise engineering, but had not previously been published or used in data modeling until now. *It is a significant advance in the discipline of data modeling.* The chapter discusses how a strategic model is defined with senior business managers in a facilitated modeling session. It includes many business examples, with case study exercise problems and sample solutions.

The second edition of this chapter incorporated extensive text changes for V3.0 of the Zachman Framework. It included a new method for the derivation from a data map of Project Maps that are used for rapid delivery of systems into production, as early milestone subproject deliverables, together with

problem exercises and sample solutions that show how this new method is applied.

Chapter 8: Strategic Alignment, Activity and Workflow Modeling, and Business Rules. An important step in enterprise architecture is strategic alignment: so that data as well as processes, locations, people, events and business plans all support each other. This chapter shows how matrices are used to achieve this alignment; these define the Governance Analysis Framework of Chapter 4. The chapter covers Activity Modeling and Activity Based Costing to define and optimize transformed business processes. It shows how to derive Workflow Models from Activity Models. It describes how Business Rules can be identified for use in these workflow models. The Business Process Management (BPM) languages discussed later in Chapter 14 use these workflow models in conjunction with Business Process Modeling Notation (BPMN), for automatic generation of executable XML-based code.

The second edition of this chapter incorporated extensive text changes for V3.0 of the Zachman Framework. It shows how activity clusters, derived from data maps using entity dependency analysis as described in Chapter 7, can be used to define activity models. It includes an important discussion of how activity based costing is used for the cost justification of enterprise architecture.

Chapter 9: Using Business Normalization for Future Business Needs. Traditional normalization (as advocated by Chris Date and others) is typically used as a technical discipline in data modeling to interview business users. This chapter describes the principles of Business Normalization: a business-driven method that is actively used by business experts and IT experts working together in a design partnership. It enables business knowledge of business experts to be used to identify future data needs for business transformation, in a way that has not yet been achieved by traditional normalization. It includes many business examples, along with case study exercise problems and sample solutions.

The second edition discussed how this chapter can also be used to study for qualification as a Certified Business Data Modeler (CBDM). It refers to the description of the Data Modeling Case Study Workshop in Chapter 6: this Workshop is the exam to qualify as a CBDM.

Chapter 10: Using Process Models to Define Business Processes. This chapter describes a Process Modeling method used to define reusable business processes. It can be used with object-oriented methods and languages, but is mostly used with the BPM languages that are described in Chapter 14.

The second edition of this chapter incorporated extensive text changes for V3.0 of the Zachman Framework. It introduced Business Process Modeling Notation (BPMN) in some detail. Process modeling using BPMN is referred to again in Chapter 14 on Business Process Execution language (BPEL). It is discussed further in Chapter 15 on modeling tools that use BPMN to define business process model diagrams. These diagrams can be automatically generated as Business Process Execution Language (BPEL) XML-based code

that is directly executed by Business Process Management (BPM) products – described in the online Chapter 14 product descriptions for the second and third editions of the book.

Technical Staff and Technology Readers

Part III: Enterprise Integration Technologies. This covers the technologies that are used for Enterprise Application Integration (EAI), Enterprise Portals, Web Services and Service Oriented Architecture (SOA).

Extensive changes were made in the second edition of Part III to cover the latest developments and products released since 2005 for Enterprise Portals, SOA, Web Services, SOA Governance, Enterprise Service Bus (ESB) and Business Process Modeling Notation (BPMN). BPMN has emerged since 2005 as an important process modeling methodology, with technology to automatically generate executable code in Business Process Execution Language (BPEL). BPM products execute this generated BPEL code for rapid delivery into production as systems. The latest versions of modeling tools that provide extensive support for UML V2.3+ and support for business process model diagrams designed using BPMN are discussed in the online chapter product descriptions for the second and third editions.

Chapter 11: Enterprise Application Integration Concepts. This chapter introduces the basic concepts of XML and EAI that are referred to and used throughout Part III. A number of software products that are offered by EAI vendors are discussed.

The latest products for EAI are included for the second and third editions, in the online Chapter 11 product descriptions.

Chapter 12: Enterprise Portal Technologies for Integration. This chapter introduces the concepts and technologies used by Enterprise Portals. It discusses their use for rapid delivery of priority information and content resources in Enterprise Integration projects.

The latest products for Enterprise Portals are included for the second and third editions, in the online Chapter 12 product descriptions.

Chapter 13: Web Services for Real-Time Integration. Web Services concepts and technologies are introduced in this chapter, along with the evolution of Web Services. It describes the technical foundations of Web Services that are used for Enterprise Integration in Part III. It discusses their use in Enterprise Portals with Web Services for Remote Portals (WSRP).

The latest developments in Web Services technologies are discussed in Chapter 13 for the second and third editions and the latest products for Web Services are included in the online Chapter 13 product descriptions.

Chapter 14: Service Oriented Architecture for Integration. The technologies used by Service Oriented Architecture (SOA) and Business

Process Management (BPM) languages are discussed. Four BPM languages are described: Business Process Execution Language for Web Services (BPEL); Web Services Choreography Interface (WSCI); Business Process Modeling Language (BPML); and Business Process Specification Schema (BPSS) for ebXML. These offer the potential to transform systems development in 21st century enterprises, with XML-based BPM languages automatically generated as executable code directly from workflow models or process models. The SOA strategies being used by vendors of EAI products, Enterprise Portals, Web Services and SOA are discussed. In 2005 Business Process Modeling Notation (BPMN) was just becoming available. The first edition therefore covered BPMN only briefly. However, BPMN and BPEL are the most significant rapid delivery technologies for the future.

After the introduction to BPMN in Chapter 10, the latest developments in BPMN, BPEL, BPML, BPSS and BPM technologies and the latest developments in SOA Governance and Enterprise Service Bus (ESB) are discussed in the second and third editions of Chapter 14. The latest products for BPM are included in the online Chapter 14 product descriptions.

Chapter 15: Managing and Delivering Enterprise Architecture Rapidly. This final chapter brings together the Methodology and Technology parts of the book. It discusses the use of modeling tools that can be used to capture the business models for Enterprise Architecture as described in Part II. It discusses several Modeling Tool vendors and briefly discusses their products in the online product descriptions. It shows how these tools can be used with the technologies discussed in Part III for rapid delivery into production of priority business processes as databases and systems needed for business transformation. It concludes with a summary of the main methodology and technology messages from the book.

The modeling tools that support BPMN are updated in the second and third editions of Chapter 15 and the latest modeling tool products for BPMN and BPM are included in the online Chapter 15 product descriptions. A discussion of Enterprise Architecture Standards is included in Chapter 15. These standards can be used as the basis for more rigorous documentation of models developed using the methods and technologies discussed in the book.

Appendix 1 : What is Wrong with Information Engineering and how to Fix it ? This Appendix has been added in the Third Edition to discuss the history of Information Engineering and the systems development breakthroughs that were achieved using the original business-driven and data-driven variant of IE that was developed in Australia in 1976 – 1980. It considers the difficulties encountered with the more widely used process-driven and IT-driven variant of IE that was developed in the 1980s by James Martin and his associated companies. The Appendix concludes with a discussion of the evolution of the enhanced version of IE, called Enterprise Engineering, for the rapid delivery of Enterprise Architecture.

Readers who have had some prior experience with IE may wish to read Appendix 1 first, and then return to Chapter 1. Otherwise, it can be read after

completing the book. The appendix then serves as a useful summary of the key messages from the book.

Enterprise Architecture Readers

Enterprise architecture readers will want to read the entire book in its published sequence to understand the business methodologies in Part I and Part II, and the rapid-delivery technologies in Part III. With this audience in mind, the book has been structured to lead you progressively through each method and technology. The end result of this emphasis on enterprise architecture is the rapid delivery of priority activities, processes, databases, and systems into production in 3-month increments. For this reason, I suggest you read the entire book from cover to cover.

Additional Materials

The first edition included a CD-ROM that was supplied with the printed book. This contained additional book material as well as student editions of several modeling tools.

The second and third editions are ebooks that provide online access to the same materials.

Online Book Materials

Additional book materials are provided online, such as questionnaire templates, case study problems and sample solutions, and product descriptions. The following Part II and Part III files are provided for download as Word or PDF documents for online access, as listed in Tables P.1 and P.2.

Table P.1 Part II Problems and Solutions Downloaded Free Online [⁸]

Part II File Name	File Contents
"Chap-03-Questionnaire.zip"	Chapter 3: Business Planning Questionnaire Template (in Word, zipped)
"Chap-06-Problems.pdf"	Chapter 6: Data Mapping Problems (in PDF)
"Chap-06-Solutions.pdf"	Chapter 6: Data Mapping Sample Solutions (in PDF)
"Chap-07-Questionnaire.zip"	Chapter 7: Strategic Modeling Questionnaire Template (in Word, zipped)
"Chap-07-Problems.pdf"	Chapter 7: Strategic Modeling Problems (in PDF)
"Chap-07-Solutions.pdf"	Chapter 7: Strategic Modeling Sample Solutions (in PDF)
"Chap-07-VSCProject.pdf"	Chapter 7: Very Small Company Project Description (in PDF)
"Chap-08-Problems.pdf"	Chapter 8: Strategic Alignment Problems (in PDF)
"Chap-08-Solutions.pdf"	Chapter 8: Strategic Alignment Sample Solutions (in PDF)
"Chap-09-Problems.pdf"	Chapter 9: Business Normalization Problems (in PDF)
"Chap-09-Solutions.pdf"	Chapter 9: Business Normalization Sample Solutions (in PDF)

⁸ The files in Table P.1 above can be downloaded from [http://www.ies.aust.com/EA_Book/\(Part-II-FileName\)](http://www.ies.aust.com/EA_Book/(Part-II-FileName)).

Table P.2 Part III Product Description Files Downloaded Free Online [9]

Part III File Name	File Contents
"Chap-11-Products.pdf"	Chapter 11: Enterprise Application Integration Product Descriptions (in PDF)
"Chap-12-Products.pdf"	Chapter 12: Enterprise Portal Product Descriptions (in PDF)
"Chap-13-Products.pdf"	Chapter 13: Web Services Product Descriptions (in PDF)
"Chap-14-Products.pdf"	Chapter 14: Service-Oriented Architecture Product Descriptions (in PDF)
"Chap-15-Products.pdf"	Chapter 15: Modeling Tools Product Descriptions (in PDF)

Format Conventions for the Book

The book is structured for ease of use both as a textbook for universities as well as for business and technical readers. Quotes are shown in italics, while Footnotes are embedded in the text: they are numbered and surrounded by [square brackets], with each numbered Footnote at the bottom of the page. Important points, as well as the contents of key tables and also bullet points in the summary at the end of each chapter are highlighted throughout the book with a light cyan background.

Part II includes questionnaires and case study problems—together with sample solutions—so you can assess your understanding of the concepts that are covered. These questionnaires, problems, and solutions are included as Word or PDF files from the relevant online links (see Table P.1).

Each chapter in Part II includes footnotes and links that also indicate the name and location of the relevant file. Figures or reports in each Problem file are prefixed with "P" (such as Figure P6.1 or Report P7.1), while figures in each Solution file are prefixed with "S" (such as Figure S6.1). All numbered footnotes for files available as Internet links appear at the bottom of each page where they are referenced.

Part III includes product descriptions for many vendors that offer products based on the technologies discussed in each technology chapter, so you can understand how the various technologies are used. These product descriptions are included as PDF files for easy downloading (see Table P.2).

Each chapter in Part III includes footnotes indicating the name and URL for further details of relevant vendors or products. Figures in each Product's PDF file are prefixed with "P" (such as Figure P13.1). All numbered footnotes for files appear at the bottom of each page where they are referenced.

Free Use of Modeling Tools

A number of modeling tools from Visible Systems Corporation [10] are supplied free, as student editions. Links are provided in the file "Chap-15-Products.pdf"

⁹ The files in Table P.2 above can be downloaded from [http://www.ies.aust.com/EA_Book/\(Part-III-FileName\)](http://www.ies.aust.com/EA_Book/(Part-III-FileName)).

¹⁰ Further information on the Visible software products is provided in Chapter 15 and is also at

(see Table P.2) so these modeling tools can be downloaded from Visible's web site. These are limited-capacity, but full-functionality versions of the following modeling tools:

- Visible Advantage Enterprise Architecture Edition
- Visible Analyst Zachman Framework Edition
- Visible Developer Code Generator
- Visible Polaris Change Management

Use the Authorization Code "991Student877" as detailed in "Chap-15-Products.pdf" [¹¹] to download the Student Editions of these products. These modeling tools are described next:

- *Visible Advantage Enterprise Architecture Edition* is a modeling tool offering powerful enterprise architecture planning and analysis support, with project examples from the book. It supports strategic planning, with integrated logical and physical data modeling, activity modeling, and process modeling. Based on a concurrent relational repository for single-user or multi-user environments, it also supports model analysis validation and automatic derivation of project plans from data models using entity dependency analysis (see Chapter 7) for full-scale enterprise architecture planning and analysis. This modeling tool has been extensively used for very large projects, such as the US Navy and the US Marines projects.
- *Visible Analyst Zachman Framework Edition* is a modeling tool that is used for enterprise architecture design and development. Models can be exported and imported between Visible Advantage and Visible Analyst. The latter also offers support also for software engineering and UML. It includes structured analysis and design modeling capabilities, object-oriented modeling, and database modeling support for forward engineering and reverse engineering. It includes model validation and uses an integrated repository for single-user or multiuser environments. The Zachman framework is used as a front-end interface for better management of repository objects.
- *Visible Developer Code Generator* is a tool for enterprise architecture deployment, with automatic code generation of Visual Basic, ASP, Visual Basic.Net, ASP.Net, and C#.Net of executable and layered applications that are customizable. Generated applications can be seamlessly and automatically connected to multiple legacy databases to deploy a common and consistent multi-tiered application framework. It generates 80% to 90% of code automatically, based on database code patterns, while managing application-specific code without change when regeneration is required.

<http://www.visible.com>.

¹¹ The PDF file at http://www.ies.aust.com/EA_Book/Chap-15-Products.pdf provides links that need Authorization Code "991Student877" to download the Student Editions.

- *Visible Polaris Change Management* is used for issues management, task management, and project management of the software development life cycle (SDLC). It is used for enterprise architecture change management. It includes task and workflow management, automated defect tracking, consolidated project information in the form of bug tracking, defect tracking, issue tracking, problem tracking, and automated ticketing. Easy to learn and use, it is configurable to enterprise architecture processes and integrates with all EA activities (planning, analysis, design, development, and deployment).

Refer to the online product descriptions for Chapter 15 for each of these products. Review also the tutorials and manuals that are provided with the downloaded products.

Using the Modeling Tools for Full-Capacity Projects

The modeling tools, code generation, and change management tools provided by Visible Systems Corporation for use with the book enable more extensive undergraduate projects to be set, based on the student edition versions that are provided. Some of these limited-capacity student edition products can be converted to the full-capacity product for use in one full-capacity project for a period of from 30 days to 90 days—at no further charge. This makes the tools invaluable for student development of larger projects for postgraduate and doctoral theses. Visit Visible Systems Corporation [¹²] and enter the Authorization Code “816EA541” to download these full-capacity versions, as described in the Chap-15-Products PDF.

In-house Courses Available for the Book

The concepts in the book are based on a series of public and in-house courses presented by the author worldwide [¹³]. These courses are delivered by PowerPoint with full instructor notes, and also include many video clips by John Zachman and Clive Finkelstein.

These presentation materials are used as the basis for easy introduction of enterprise architecture methods and technologies to your educational curriculum. Each course contains sections that correspond to specific chapters of the book. They provide more detail than can be included in the book. While new developments will be published as separate editions of the book approximately every 5 years, each course is continually maintained with the latest, up-to-date methodology and technology developments.

¹² The PDF file at http://www.ies.aust.com/EA_Book/Chap-15-Products.pdf provides links for the download versions that need Authorization Code “816EA541” for full-capacity projects.

¹³ Course outlines of some public and in-house courses that are presented by Clive Finkelstein worldwide are available at <http://www.ies.aust.com>. Course outlines accessible from the *Courses* link of any page present concepts suitable at the information systems and business school undergraduate level. Project descriptions accessible from the *Projects* link of any page also cover concepts suitable for postgraduate and doctoral levels.

Options for certification are also available, if required. For example, the Certified Business Data Modeler (CBDM) self-study course is based on Chapters 6 and 9 of this book. It includes the Data Modeling Case Study Workshop. This Workshop is the CBDM qualification Exam [¹⁴].

Availability of HD Video Courses based on the Book

The material covered in this book is also available as two High Definition video courses: “*Rapid Delivery Workshop for Enterprise Architecture*”, which covers Parts I and II and “*Web Services and Technologies for Enterprise Architecture*”, which covers Part III. These courses were developed for use in Universities and large government and commercial organizations. The author, Clive Finkelstein, presents the two courses over 40 hours using this book as the textbook. Details are available from Visible Systems Corporation [¹⁵].

University and Corporate Use of the Book

The book and the student edition modeling tools discussed above are for use by universities and other educational institutions for undergraduate and postgraduate courses. They can also be used by large commercial, government, and defense organizations for internal training of business and IT staff. The educational materials provided use this book as a comprehensive reference textbook and use the modeling tools described above for strategic planning, data modeling, activity modeling, process modeling, and object-oriented modeling in UML; for automatic code generation in VB, ASP, VB .Net, ASP .Net, and C# .Net; and for change management.

Structured Chapters for Rapid Delivery

Each chapter in the book has been designed to stand alone, if required. It covers a specific methodology or technology for rapid delivery of enterprise architecture. The book, however, is intended be used in its entirety: All methods and technologies in the chapters are presented in the recommended sequence for rapid delivery of enterprise architecture.

Each chapter covers introductory concepts, with increasing detail as the student works through each topic. For the methodology chapters in Parts I and II, case study problems and sample solutions enable each student’s understanding to be tested. For the technology chapters in Part III, product descriptions and vendor strategies are discussed. Each chapter summary highlights the key principles that have been learned. Product descriptions are included as links for online download as PDFs.

¹⁴ Further information and pricing for the CBDM self-study course is also available from the IES web site at <http://www.ies.aust.com/>.

¹⁵ The two HD Video Courses by Clive Finkelstein: “*Rapid Delivery Workshop for Enterprise Architecture*” and “*Web Services and Technologies for Enterprise Architecture*” that both use this book as textbook are available from Visible Systems Corporation at <http://www.visible.com/>.

Copyright and Trademark Acknowledgments

All product names and all registered and other trademarks that appear in this book are, and remain, the property of their respective owners. They have been included for reference purposes only. Any further information about any product or service referenced in this book should be obtained from the relevant product or trademark owner, based on the links supplied in the footnotes at the bottom of each page, or other links that are obtained through appropriate Internet searches.

Other Acknowledgments

I would like to acknowledge the input and suggestions provided by the following people:

- John Zachman, president of Zachman International. Many thanks for writing the foreword, and for his extensive support and suggestions in relation to Chapters 1 and 5, and elsewhere.
- Thanks to Joe Butchko (U.S. Air Force, Retired) for his assistance in Chapter 5 on the use of enterprise architecture at the air mobility command of the U.S. Air Force.
- Thanks to Bob Weisman, when he was enterprise architecture practice manager of CGI in Ottawa, Canada, for his input to Chapter 5 on FEAF in federal government, as well as the use of enterprise architecture, C4ISR and DoDAF in the defense forces of the United States, Canada, the United Kingdom, and NATO. Thanks also go to Bob for his technical review of the entire book.
- Stan Locke, previously managing director of Zachman Framework Associates. Many thanks for his support and suggestions in relation to Chapters 1 and 5, and elsewhere.
- Thanks to Doug Erickson for his input to Chapter 5 on his project experience at the Ohio Bureau of Workers' Compensation.
- Thanks to Mike Tiemann for his assistance with the enterprise architecture planning (EAP) approach used by Steven Spewak, referenced in Chapter 5.
- Many thanks also to Ram Kumar in Sydney for his review of Chapters 13 and 14 and his suggestions for improvement based on his role as an OASIS working group chairman.
- I want to thank George Atallah, IBM federal systems in Washington, D.C., for his input and red book references on virtualization and on-demand directions in Chapter 15.
- Many thanks also to George Cagliuso, Chairman and Mike Cesino,

President and CEO of Visible Systems Corporation, for making the limited-capacity student edition versions of their products: *Visible Advantage*, *Visible Analyst*, *Visible Developer*, and *Visible Polaris* available to the readers of this book and for also allowing some of these products to be used for a period of from 30 days to 90 days for one full-capacity project.

