Preface

The vast majority of enterprise Information Systems (IS) were implemented in the early days of computing by using the traditional paradigms and implemented with structured computer languages such as COBOL. These legacy systems are standalone systems that are limited in scope as they tend to be infrastructure-specific, not integrated with other systems, and usually, information is not readily shared between systems.

With recent developments, particularly the trends towards e-Commerce, Enterprise Resource Planning (ERP), and Supply chain management (SCM), many companies are realizing that they will have to migrate to these new systems in order to remain competitive. The migration from legacy systems to modern systems could be challenging from the business and technical point of view. However, such projects often failed for several reasons:

- the new systems were quite expensive,
- cumbersome to implement,
- involve radical or unwanted changes to existing processes,
- resistance to change,
- sharing sensitive information among departments and companies,
- vulnerable to security threats,
- requires extensive training, et cetera.

Each modern system has its own set of challenges when it comes to the migration of legacy systems.

The migration of legacy systems into ERPs is challenging as these are integrated pre-packaged systems designed to support many organizational functions and their customization is normally discouraged. In order to reduce the risk of failure for implementation, vendors encourage business process re-engineering, rather than software customization of ERPs. It is often argued that it is more advantageous for companies to use them without changes as these were built based on best practices that incorporate industry standards. Vendors encourage business process reengineering not only to be able to easily implement ERPs but also in order to achieve dramatic improvements in one or more performance measures in the business that is adopting the ERP. Business process reengineering can increase the risk of failure of a project due to resistance to change from organizations, difficulty in training employees, and business requirements becoming obsolete, and these risks need to be managed properly during the project execution in order to maximize success rates.

The reengineering of legacy systems to E-Commerce can also be a challenge as this process relies heavily on the ability to integrate them with web applications and web services such as payment systems.

For these, legacy systems would need to support Web architectures such as the Service Oriented Architecture. This support would require a change of paradigm from structure to component based systems that would require a change of software models from structured models such as data flow diagrams to UML based models.

Supply Chain Management systems are normally used to integrate business functions and business processes within and across companies, into a cohesive and high-performing business model. Legacy systems need to be reengineered to support modern supply chain management systems such as e-procurement. Some concerns for the reengineering of legacy systems into supply chain management systems include security of data exchanged, integration of business processes among suppliers, data exchange over networks, transformation of legacy systems into Web services, and information flow among different supplier tiers in the supply chain.

Although each modern system might require a different form of legacy systems reengineering, information systems reengineering in general has the objective of extracting the contents, data structures, and flow of data and process contained within existing legacy systems in order to reconstitute in to a new form and subsequent implementation. Businesses must constantly adapt to a dynamically changing environment that requires choosing an adaptive and dynamic information architecture that has the flexibility to support both changes in the business environment and changes in technology.

The book covers different techniques that could be used in industry in order to reengineer business processes and the legacy systems into more flexible systems capable of supporting modern trends such as ERP, Supply chain management systems, and E-commerce. The book also covers related aspects to the reengineering of legacy systems: risk management and obsolescence management of requirements.

The book consists of thirteen chapters, each focusing on reengineering of legacy systems to turn them into modern enterprise systems.

Chapter 1 presents an approach to improve business agility of a legacy IT system by modernizing COBOL application development. The special features in this approach include the introduction of service oriented architecture (SOA) for Web application integration, using model driven development approach with code generation, and agile development process. The legacy systems are often designed using the structured systems analysis and design (SSAD), whereas the modern systems are object oriented using the Unified Modeling Language (UML) documentation.

Chapter 2 highlights the need for converting traditional documentation of legacy systems to object oriented approach, and presents a set of rules to automate the conversion of systems, which were originally modeled using structured techniques to UML.

Chapter 3 discusses the details of systems development and evolution models mainly aiming at an ongoing reengineering of legacy systems, and proposes few strategies for reengineering of both data oriented model and process oriented models. The chapter focuses on legacy systems to incorporate the interfaces to external systems for automatically updating the data, and creating an extended process by including the supplier and customer processes.

Chapter 4 envisions an Information System as a set of interdependent components that provide the intended services, and presents a methodology for component based modeling and development of an information system, starting from the requirements definition phase, arriving at candidate components and creation of final components and their interfaces. The methodology aims at clarifying the intricate details and usage of an information system via business type models and use case models. The chapter proposes a methodology for component reengineering; model-view-control framework for component

evolution, refinement, and replacement to achieve a reengineered Information System that reflects current requirements in business domain.

Chapter 5 focuses on inter-organizational processes and identifies eight process issues that need to be taken into account when evaluating inter-organizational integration configurations. It also presents few examples upon how problems surface in a completely centralized and in a completely decentralized inter-organizational process integration scenarios.

Chapter 6 highlights the need for understanding the requirements obsolescence that differs from business to business, project to project, and also from the stakeholders' point of view to the business users' perspective involved in a project, which is mainly due to changes in business reasoning or business processes. The chapter presents a comprehensive questionnaire with relevant and probing questions that have collected data on obsolescence of requirements, in order to gather data for indentifying risk factors and their impact. This research helps in turning IT project failures into project successes.

Chapter 7 aims at automation of a business process via workflow management system or ERP system embedding workflow functionalities. The wider diffusion of ERP systems tends to favor the latter solution. The chapter reports a study assessing the "workflow ability" of ERP systems and comparing this with that of Workflow Management Systems. Then, an empirical study was conducted regarding two different case studies. The correctness and completeness of the process models implemented using ERP and WfM systems were evaluated and analyzed.

Chapter 8 emphasizes the importance of evaluating the risks while planning and implementation of ERP systems. The chapter presents a case study example focusing on the implementation phase of ERP system, stressing upon the validity of specific risk management model. It covers a brief overview of two ERP projects risk management models; the first model deals with ERP as technology where risks are managed in accordance to Information Technology and Information System (IT & IS) projects governance, while the second model considers the integrated and interdisciplinary nature of ERP systems and introduces the term "ERP Governance" as a model that demand balances between IT and business governance. A third model similar to the second is also proposed, which relies heavily on best-practises and assessment frameworks from the industry.

Chapter 9 explores and assesses the Critical Success Factors (CSF) that influence the implementation of an ERP system. The chapter investigates how many CSFs are strongly correlated with each other for the success of ERP projects in the manufacturing sector. The chapter also tests empirically using the Statistical Package for Social Science Analysis of Moment on Structures (SPSS AMOS 18.0) to justify the level of CSFs among the local and joint-venture companies using a t-test analysis.

Chapter 10 describes a debacle that occurred when a large assembler of fast moving consumer electronics in a newly formed supply chain involving four key players in a sell-buy relationship. The chapter explored mistrust of trading partners as the companies receiving component sets refused liability for damage or defects introduced by upstream companies in the supply chain. A remedial *quick-fix* using centralized inspection at the principal supplier soon was adopted to facilitate supply of complete sets of mechanical parts to the assembler. The chapter highlights significant similarities between the case study supply chain and the concepts used in business process reengineering.

Chapter 11 proposes a framework-based strategy using .net for easier migration from legacy software systems to Web services, and presents software metrics observed during the process of reverse engineering of legacy systems for designing the Web services.

Chapter 12 covers the work to produce an architecture that shows how to reengineer traditional point of sales (POS) terminal payments to adapt payments over the Internet via Web services. In current

environment of global economy, vendors can negotiate about services and fees with payment providers worldwide.

Finally, chapter 13 highlights the importance of software management, especially in Aerospace Industry, and stresses that negligence of any maintenance components can put the organization into risk. Standardizing processes are necessary to avoid or minimize the risks. The chapter looks into a reengineered process to deploy the upgrades proactively in a cost effective manner.

In conclusion, the chapters covered contemporary approaches to legacy systems reengineering into modern enterprise Information Systems, namely ERP, SCM, and E-Commerce systems. The brand new off-the-shelf ERP, SCM, or E-Commerce system requires extensive customization that often goes wrong one way or the other, and mandates extensive training to already resisting internal process owners and technical/operation staff members. The reengineering effort of a legacy system can be welcoming by the process owners as the impact to technical/operation staff is almost negligible, less expensive as the system is already well understood, and reflects upon the business processes more accurately than any new enterprise system. Therefore, the book highlights the need for reengineering a legacy systems to turn it into a contemporary enterprise system that accomplishes the same or better results than any other off-the-shelf system.

Raul Valverde
Concordia University, Canada

Malleswara Rao Talla
Concordia University, Canada