New Dog, Old Tricks: ERP and the Systems Development Life Cycle

Richard T. Grenci
Bradley Z. Hull
Department of Management, Marketing, and Logistics
John Carroll University
University Heights, Ohio 44118
bzhull@jcu.edu

ABSTRACT

This paper presents and analyzes an approach for using the systems development life cycle (SDLC) to teach enterprise resource planning (ERP) implementation issues. Not only does the SDLC put ERP into perspective, but ERP implementation issues give a substantive and informative context to the SDLC. A review of the literature provides a basis for using the SDLC as a framework for evaluating ERP implementation success and failure. In turn, the successes and failures provide a rich and interesting venue for introducing students to the relevance and implications of ERP applications as well as the SDLC. Such an introduction can be employed as a value-added teaching component in practically any IS curriculum, regardless of whether or not the institution has access to ERP software. Furthermore, the component can be used in a variety of information systems and management classes, including introduction to IS, introduction to ERP, systems analysis and design, project management, and even an MBA-level IS class. The development and analysis of the teaching approach is based on the experience of having employed the approach in an introductory ERP class. The experience reveals lessons learned, and it provides a source of data for mapping numerous ERP implementation failures to the stages of the SDLC.

Keywords: Enterprise Resource Planning, Systems Development Life Cycle, Teaching Case Study, Information Systems Curriculum

1. INTRODUCTION

In the midst of the meteoric rise of Enterprise Resource Planning (ERP) applications, the leading ERP vendors (first SAP in 1996 and then others soon afterwards) established academic alliance programs to bring their software into university classrooms (Bradford et al., 2003). Several years later, there are concerns that "many universities have struggled with how to implement such major changes to their courses and curriculum" (David et al., 2003, p. 417). Some have called for the development and broad dissemination of ERP-related classroom materials and exercises that could benefit both alliance and non-alliance faculty alike. Others have suggested that incremental integration might precede a more comprehensive, cross-functional employment of ERP software in a curriculum (Watson and Schneider, 1998). Both of these positions support the merits of creating and incorporating ERP assignments as building blocks of curricular change.

Perhaps more than most disciplines, information systems (IS) curricula are driven to change by the almost continuous innovations in information technology (IT).

However, the need for such change must be weighed against the need for curricula that "balances tradition with innovation" (Landry et al., 2003, p. 117). In other words, universities must bring new innovations and technologies into the classroom within the framework of traditional and foundational concepts that provide a knowledge base for the discipline. In the case of the IS discipline, tradition can be found in the systems development life cycle (SDLC). Specifically, a study of the knowledge areas of various IS programs revealed the SDLC to be a common curricular theme that, as a guiding framework, "highlights IT's relationship to traditional computing disciplines (Landry et al., 2003, p. 117). As such, the SDLC can be used as a framework for highlighting the relationship between ERP technology and traditional computing.

This paper heeds the call for disseminating ERP-related educational materials by presenting and analyzing an approach for using the SDLC to teach ERP implementation issues, and vice versa. Not only does the SDLC put ERP into perspective, but ERP implementation issues give a substantive and informative context to the SDLC. The approach presented here can be employed in several

different courses such as introduction to IS, introduction to ERP, systems analysis and design, project management, and even an MBA-level IS course. Although this approach is not limited to institutions with ERP software, the importance of ERP implementation issues has been highlighted by the explicit coverage of such issues in the introductory IS and ERP courses of well-known SAP University Alliance members (e.g., Stewart et al., 1999; Watson and Schneider, 1998). In addition, a review of the literature provides a basis for using the SDLC as a framework for evaluating ERP implementation success and failure. Ultimately, the successes and failures provide a rich and interesting venue for introducing students to the relevance and implications of ERP applications as well as the SDLC.

2. BASIS FOR THE FRAMEWORK

2.1 Systems Development Life Cycle and Methodologies
Information systems development methodologies (ISDM) have been employed by organizations for more than three decades. In that time, ISDMs have evolved – and continue to evolve – into various approaches and techniques (Vickers, 1999), with some estimates placing the number of methodologies at more than 1,000 (Jayaratna, 1994). With such variation, it is easy to lose sight of the defining nature of a methodology. Traditionally defined, an ISDM is the structured set of methods that are used for developing and implementing information systems applications (Blackwell, 1997). In this vein, methodologies can be differentiated from and categorized into several different conceptual approaches (Iivari et al., 2000), including (but not limited to) structured, object-oriented, data-oriented, and user-centered. Other categorizations (e.g., Avison and Fitzgerald, 2003) often include prototyping as a definitive approach as well. Each approach can be (and many have been) operationalized into numerous specific methodologies that dictate very specific sets of tasks and deliverables.

While ISDMs may differ in concept and technique, many employ similar types of tasks that often are described with respect to the stages of a systems development life cycle. For example, the stages of the object-oriented (OO) systems life cycle can be mapped to those of the structured SDLC (Henderson-Sellers and Edwards, 1990). Likewise, a prototyping approach, which has been characterized as a methodology as well as an analysis and design technique (Blackwell, 1997), also can share the same stages as the structured SDLC (Plyler and Kim, 1993). Comparisons to the structured approach are due to the fact that the traditional systems development methodology – also known as the waterfall model – explicitly identified and followed a linear sequence of life cycle steps and thus became associated with the definition of SDLC. According to one review of the typical project development stages and activities (which have been reformatted for Table 1), such a structured approach was "introduced in the 1960's by the UK National Computing Centre" (Management Outline, 1998). In addition to project development activities, the life cycle continues with the *post hoc* problem identification of new business needs, which begins the cycle again.

2.2 ERP Implementation Life Cycle
Like any other systems development effort, an ERP implementation also can be assisted by a development methodology. Perhaps the best known ERP-related methodology is SAP AG's Accelerated SAP solution, or ASAP. Actually, SAP describes ASAP as a full life cycle solution that includes the ASAP Roadmap methodology. Table 2, based on information provided in SAP's Help Portal (help.SAP.com), lists the five phases and associated activities of Roadmap. Consistent with the structured SDLC, the Roadmap phases likewise continue beyond development activities to include ongoing continuous improvement that is driven by changes in technology or business processes.

Table 1: Typical SDLC Stages and Activities (as described by Management Outline, 1998)

Life Cycle Stage	*Activities*	
Feasibility Study	- examine current system - identify problems - identify solutions	- evaluate costs/benefits - recommend solution - set goals for system
Systems Investigation And Analysis	- analyze old/new system - analyze processes	- analyze functionality - analyze users
Systems Design	- overview functional elements - detail inputs/outputs - detail screen/interface design	- hardware requirements - operating instructions - testing/changeover procedures
Implementation	- build/program system - test system and fix errors - install hardware	- train users - change over to new system - support new system

278

Table 2: ERP Life Cycle: ASAP Roadmap (as described at help.SAP.com)

Life Cycle Stage	*Activities*	
Project Preparation	- initial project planning - project procedures	- project kickoff - technical requirements
Business Blueprint	- develop system environment - define organization structure	- business process analysis - business process definition
Realization	- baseline configuration - final configuration	- develop programs, interfaces - final integration test
Final Preparation	- stress and volume tests - training	- cutover
Go Live and Support	- production support	- performance review

Table 3: Comparability of SDLC and ERP Life Cycle

Focus of SDLC Stage	*Focus of ERP Stage*	*Common Types of Focal Points and Activities*	
Planning	Planning	- alternatives - feasibility - cost/benefit	- solution selection - goal setting - project initiation
Analysis	Analysis	- process analysis - user analysis - data analysis	- process redesign - requirements definition - system interfaces
Design	Configuration	- inputs, outputs - screens/interface - modules/integration	- prototyping - customization - configuration
Implementation	Installation	- programming - installation - testing	- training - change over - troubleshooting
Support	Support	- support - review	- fixing - modifying

For the most part, the phases of the ASAP Roadmap are consistent with those of the structured SDLC. In fact, traditional and ERP life cycles have been compared or contrasted in studies that advance ERP implementation methodologies. One recent study (Ahituv et al., 2002) defined phases of the ERP life cycle based on an integration of the traditional SDLC, prototyping, and application package development models. Although some of the specific tasks differed between models, the general phases (selection, definition, implementation, and operation) were comparable to the structured SDLC phases. Another study (Parr and Shanks, 2000) defined a model of ERP implementations based on three major project phases (the second of which included five sub-phases) as follows: 1. planning; 2. project (which includes set-up, reengineering, design, configuration and testing, and installation); and 3. enhancement. Ultimately, the SDLC and ERP life cycle stages are broadly comparable and employ common types of activities.

Table 3 draws a parallel between the stages and activities of the traditional SDLC (listed in Table 1) and the ERP life cycle (listed in Table 2). As shown in Table 3, both life cycles focuses on planning and analysis in the first two stages, and support in the last stage. In addition, SDLC design activities are comparable to ERP configuration activities; and SDLC implementation activities are comparable to ERP installation activities. Ultimately, both life cycles share common stages and a common set of focal points and activities.

2.3 ERP Implementation Success Factors
The purpose of an IS methodology is to help to ensure the successful development or implementation of a software application. However, some claims place failure rates for ERP implementations at 40 to 60% (Langenwalter, 2000). Researchers and practitioners alike have studied and advanced the understanding of ERP success factors. Such advancement is relative to two questions: 1. what is success

(a question that is relevant to goals or expected benefits); and 2. how is it achieved (a question that is relevant to key success factors)?

One study (Umble et al., 2003) summarized ERP success factors into ten categories related to: goal setting, executive commitment, project management, organizational commitment, project team, training, data accuracy, organizational change, multi-site issues, and technical difficulties. Another study (Mabert et al., 2003) defined thirty ERP success factors, but grouped them into three general categories of planning, design (identified as "implementation decision"), and implementation. Relevant to the research at hand, these three categories are comparable to ERP life cycle stages. In fact, a closer look at the variables could allow for some of them to be broken into a fourth category that corresponds to the analysis stage, leaving the support stage as the only omission. Furthermore, the same study (Mabert et al., 2003) also considered the impact of the success factors on project goals related to schedule and budget. This provides for an informative perspective as to when and how an ERP implementation succeeded or failed based on the stage of the ERP life cycle.

2.4 Evaluating ERP Success and Failure
The goals and successes of ERP implementations extend well beyond project schedules and budgets to include numerous intended organizational benefits. A recent survey by Metadata produced a top ten list of intended accomplishments that included improvements in: software, financials, analytics, standardization, performance, service, systems, purchasing, ordering, and personnel costs (Fox, 2003). At a higher level, categories of ERP benefits can be organized into general groups, such as: operational, managerial, strategic, technological, and organizational (Al-Mashari et al., 2003). With wide-ranging impacts, the overall success or failure of an ERP implementation may be difficult to definitively establish.

As was previously noted, one way to evaluate the success or failure of a development project is to consider it on a life-cycle, stage-by-stage basis. This method is similar to and is complemented by a narrative approach to evaluating success and failure. In contrast to a rationalist approach of evaluating fixed expectations and outcomes, the narrative approach employs a more descriptive method of documenting development issues so as to place the entire project into perspective (Fincham, 2002). As such, the progress and outcomes of a project can be placed into a more evolutionary context. In fact, using a similar basis of the benefits of narrative, case studies have been promoted as an effective pedagogical tool for engaging students in the concepts of information systems development (Ramiller, 2003). Ultimately, by using the SDLC as a framework, an incremental and narrative evaluation of success and failure can be systematically directed towards a comprehensive and engaging analysis of ERP implementations.

3. TEACHING APPROACH

Using the SDLC as a framework to evaluate the success and failure of ERP implementations, a teaching approach has been developed by the corresponding author of this paper to introduce students to ERP applications and issues. The approach – referred to as "Welcome to ERP" – has been used for the past four years at John Carroll University, a Jesuit institution of 3,000 students located in Cleveland Ohio. John Carroll provides a liberal arts education in which students declare their majors (including business majors) just prior to their junior year. As such, most business majors take the required business IS course during their junior year. Business Information Systems (BIS) majors would go on to take additional BIS courses (including systems analysis and design and project management), but an elective ERP class is taken by a variety of business majors, primarily during their senior year. It is in this elective ERP class where the teaching approach has been developed and used.

"Welcome to ERP" is centered on a one-week project in which students are divided into groups and each group investigates one of the better known ERP failures. By sharing their knowledge through presentation and discussion, students self-discover the basics of ERP and gain a fuller appreciation of ERP implementation issues as well as the systems development life cycle. Although the SDLC-based approach can be used quite effectively as a stand-alone learning experience, "Welcome to ERP" includes a preceding component to introduce ERP basics via a one-week class discussion of the book *Why ERP?* by Jacobs and Whybark. Furthermore, since John Carroll is a member of the SAP University Alliance, "Welcome to ERP" also employs a one-week follow-up component involving hands-on exposure to ERP software. However, the SDLC-based approach would work equally well in universities that do not have access to ERP software. For one thing, even absent of hands-on exercises, the approach still provides for a contextual introduction to ERP applications. In addition, a hands-on component can be easily provided by computer-based training (CBT) exercises that do not require access to an ERP software installation.

The following sections discuss the three interactive exercises, collectively called "Welcome to ERP." Each exercise lasts approximately one week. They are: 1. a case analysis of an ERP novel, 2. student presentations of ERP failures, and 3. some ERP hands on exercises either with a live ERP system or CBT courses.

3.1 Step 1: Case Analysis of an ERP Novel
As a practical introduction to the topic, students are assigned to read *Why ERP? A Primer on SAP Implementation*, by Jacobs and Whybark. This excellent 127-page novel is written along the lines of *The Goal*, by Eliyahu Goldratt. While it is assigned and used as a case study, students find it easy and fun to read. The hero is Billy Armbruster, the operations manager of a custom build

furniture company. Billy finds out that his company is likely to be involved in a major IT project (an SAP installation) and he is assigned to investigate and make recommendations to his boss. He investigates SAP, recommends against using it, is forced to implement it, quits, and finally joins an SAP consulting group.

From this brief assignment, students learn many of the basics, such as: what is enterprise software, what are its pros and cons, what is the role of an ERP consultant, and what are some of the implementation problems? The book clearly shows the importance of committed management (which later is revisited as a critical aspect of the systems development life cycle). These points are reinforced in the class discussion, led in part by questions such as:

1) Why did Billy's management decide to install SAP? Were those good reasons?
2) Can SAP work for the company, a small manufacturer of highly customized products?
3) Do you think that SAP would make the company more profitable?
4) Did management contribute to the eventual failure? How?
5) What resulted from the SAP implementation?
6) Billy joined an SAP consulting company. What is the consulting company's role in SAP implementation projects? Why not install SAP yourself?

The book also has a website which offers other suggestions for leading the discussion.

The book and discussion help students appreciate the issues by putting them in Billy's shoes to gain a first-hand view of the problems that he encounters. The case format allows the instructor to expand on the individual topics as they emerge in class discussion. In addition, it leaves the students with an understanding that any IT project must be planned carefully and well in advance because it can go wrong at many points. It impresses on students that implementation is a multi-step process, and that management and consultants play an important role. In turn, these issues justify using the SDLC as a relevant analytical framework for evaluating ERP implementations.

3.2 Step 2: Student Presentations of ERP Failures

Once the students have a basic appreciation of ERP implementations, both depth and breadth are added to their contextual knowledge, again by way of exploration. This is accomplished with student group projects based on an SDLC analysis of ERP implementation failures (see Table 4 for a description of the project). A list of well-known ERP failures is provided along with the project description. These failures could be left for students to find in IT mass media articles, but several years of utilizing this project has allowed for a detailed compilation of a list of failures. By categorizing the failures based on the types of problems and the SDLC stages where the problems occurred, this list allows the instructor to hand pick particular cases so as to guide the students towards a range of problems. Each group of students selects from among the well-known ERP failures and then researches and analyzes that particular

failure. During the week that they are doing the research, students are given a handout that describes the SDLC. In addition, students may be directed towards more formal readings about ERP (e.g., Davenport, 1998), thus motivating them to use a range of resources to support their research.

The spirit of the project is for students to get to the bottom of their particular problem, and then to compare the problems researched by the other groups.

As such, there is a benefit to providing a directed list of failures for students to choose among so as to cover a range of implementation issues that can be highlighted in the student presentations. To that end, Table 5 presents a list of failures that is organized based on the type of problems, the SDLC stage where the problems occurred, and the companies that faced those particular problems in those particular stages. The comprehensive list and categorization is designed for the use of the instructor; therefore, it is not shown to the students who are meant to discover the problems themselves through their research.

At the end of each presentation, the instructor builds a spreadsheet that shows the researched companies and the reasons for failure, based on the findings of each team. The students have a great interest in contributing to the development of the spreadsheet as they begin to see more and more ways a project can fail in comparison to the ERP implementations researched by the other groups. The success of this project stems from the fact that students reach various conclusions themselves, rather than through lectures. Learning comes from their involvement in their presentation and how their company's failure differed from the others.

Several points are revealed when developing the spreadsheet:
1) There is rarely a single reason for failure. Usually many things contribute to it. Also, failure at one step contributes to failure at the successive steps.
2) The importance of management commitment is critical. Lack of commitment appears as a major cause in most of the failures.
3) Poor testing is often the culprit. Testing can be seen as an impediment to going live, and is often overlooked, especially if the deadline is looming and the project is late.
4) A big bang can create enormous problems. Many of the large failures on the list were big bangs.
5) When failure occurred, the company typically buckled down and made it work. Quitting was not an option. Most of these companies are now successful and wiser ERP users.

Other points can be emphasized while discussing the student presentations:
1) Worldwide there have been 30,000 ERP installations. Many have been problematic but problems are a fact of life. But many have been well executed. In this

Table 4: Student Group Project: Analysis of ERP Failure

Project Statement:
Select a company that experienced a major ERP problem, find out what the problem was, what they did about it, and what they are doing today.

Project Deliverables:
1. A 20 minute class presentation describing what the company does, what problems they experienced, how the company responded, and what they are doing today. Did they stay with the same software and make it work, or did they abandon ship?
2. A 2-3 page paper succinctly summarizing the problems identified.
3. URLs or hard copy of five highly relevant articles on their failures.

Students may use the Internet, library sources, and any other means of investigating the failure. Below are some technology search engines and online resources to get you started:

www.baselinemag.com	www.cfo.com	www.cio.com
www.computerworld.com	www.darwinmag.com	www.datamation.com
www.informationweek.com	www.isworld.org	www.ittoolbox.com
www.itworld.com	www.line56.com	www.manufacturing.net/pur,
www.midrangeenterprise.com	searchcio.techtarget.com	www.serverworldmagazine.com

project, they are only looking at the exceptions.
2) Even the companies that declared bankruptcy use ERP systems today. Fox Meyer was purchased by McKesson, an SAP shop; Tri Valley Growers was purchased by Del Monte, a Baan shop; LTV Steel was reformed as ISG, a user of a second-tier ERP.
3) Students located their facts through the IT media, but the real truth may be somewhat different from what they read.
4) What are some alternatives to big bang implementations?
5) The importance of the Systems Development Life Cycle.

The project wraps up in two ways, beginning with a formal run through of the systems development life cycle to reinforce it. As seen in Table 5, the problems experienced can be mapped to the SDLC, making it a useful concept. Second, it concludes with a discussion of the extensive industry survey by Mabert et al. (2003); and the findings of the survey are compared to the results of the student presentations. For example, Mabert et al. conclude that large companies tend to avoid big bang implementations. This contrasts with the student results, which were very likely among the few big companies that attempted a big bang. Also, Mabert et al. discuss the differing ERP benefits experienced by large versus small companies, and the degree of customization in each. Examining several such topics places the student results into perspective.

3.3 Step 3: Hands-on ERP Exercise
The final one-week component of the teaching approach is to provide some exposure to ERP software. As an SAP University Alliance member, John Carroll University has access to both SAP and SAP's practice database for a company called IDES. The students are given hands-on experience in three areas. First they process an order through to invoicing, so that they experience a business process. Second, they locate answers to business questions by searching the IDES database, so they see the importance of centralized data. And third, they explore some of the "drilldown" capabilities of the Logistics Information System, emphasizing the decision support features of SAP. Universities with other ERP systems can provide similar exercises.

Universities without ERP systems can experience the functionality of an ERP application through several good CBT courses. In particular, NetG offers tutorials which guide the student through SAP screens at their own pace, without the need for a live system. This is sufficient to provide them with an "ERP experience." In fact, John Carroll has very successfully used NetG tutorials even in advanced ERP courses that had access to SAP software. The tutorials allow students to quickly master the manual interface aspects of SAP along with many of the primary business processes contained in each SAP module. Classroom lectures then can focus entirely on more conceptual aspects. A good analogy can be made with respect to courses using Microsoft Excel. Students can master the manual interface outside of class using tutorial material, and classroom discussions can show how to use Excel to solve problems.

Other points can be emphasized while discussing the student presentations:
1) Worldwide there have been 30,000 ERP installations. Many have been problematic but problems are a fact of life. But many have been well executed. In this project, they are only looking at the exceptions.

Table 5: Categorizing Companies' ERP Failures by SDLC Stage Where Problem Occurred
(Useful as Instructor's Guide to Selecting ERP Failures for Student Projects)

SDLC PROBLEMS BY STAGE	COMPANIES (MENTIONED IN IT MASS MEDIA) THAT EXPERIENCED THESE PROBLEMS
Planning Problems	
- big bang caused big problems	Fox Meyer, Hershey, Nike, Snap On, and many others
- overly ambitious goals	Fox Meyer, Hershey, Nike
- did not involve business units	Fox Meyer, Nestle's
- busy season "go live"	Greyhound, Hershey's, Whirlpool
- shortened the "go live" date	Fox Meyer, Grainger, Greyhound, Hershey's
- business changed during project	Fox Meyer, Grainger, Greyhound, Snap-On
- lack of commitment, leadership	Aquatech, Greyhound, Hershey, Nestle, Nike, Snap On, Whirlpool
Analysis Problems	
- complicated requirements	Allied Waste, Nestle, Snap-On
- many packages at once	Fox Meyer, Hershey, Nestle, Nike
- requirements errors	W.L.Gore
- inadequate process analysis	Fox Meyer, Greyhound, Hershey, Nestle, Snap On
- too little time allowed	Cleveland State University (CSU), Fox Meyer, Whirlpool
Design Problems	
- too little capacity	CSU, Dell, FoxMeyer, Greyhound, Whirlpool
- software bugs, bad functionality	CSU, Nike, Snap-On
- used a beta version	CSU
- high degree of customization	Allied Signal, Jo Ann, Nike, Volkswagen
Implementation Problems	
- didn't heed vendor warnings	CSU, Fox Meyer, Grainger, Greyhound, Nike, Whirlpool
Configuration	
- inexperienced consultants	Fox Meyer, Hershey, W.L.Gore, LTV Steel
- multiple consultants	Hershey, LTV Steel
- no consultants - diy	CSU, Nike, Greyhound
- configuration mistakes	Goodyear
Integration	
- problems integrating with other systems	Hershey, Nestle, Petsmart, Tri-Valley
- problems integrating between modules	Nestle, Jo Ann
Testing	
- minimal or no testing	Fox Meyer, W.L. Gore, Hershey, Nike, Goodyear
- went live after testing failed	Fox Meyer, Greyhound, Whirlpool, CSU
- went live using test data	W.L.Gore, Norfolk Southern
Support Problems	
- poor employee training	Fox Meyer, Grainger, Hershey, Nestle, Snap On
- worker rebellion	Nestle
- eliminated experienced employees	Greyhound
- much data not entered	Hershey
Big Problems	
- ERP Related Bankruptcies	Fox Meyer, LTV Steel, Tri Valley Growers
- ERP Project Cancellations	Aquatech, Air Products, Dell, Starbucks

2) Even the companies that declared bankruptcy use ERP systems today. Fox Meyer was purchased by McKesson, an SAP shop; Tri Valley Growers was purchased by Del Monte, a Baan shop; LTV Steel was reformed as ISG, a user of a second-tier ERP.

3) Students located their facts through the IT media, but the real truth may be somewhat different from what they read.

4) What are some alternatives to big implementations?

5) The importance of the Systems Development Life Cycle.

The project wraps up in two ways, beginning with a formal run through of the systems development life cycle to reinforce it. As seen in Table 5, the problems experienced can be mapped to the SDLC, making it a useful concept. Second, it concludes with a discussion of the extensive industry survey by Mabert et al. (2003); and the findings of the survey are compared to the results of the student

presentations. For example, Mabert et al. conclude that large companies tend to avoid big bang implementations. This contrasts with the student results, which were very likely among the few big companies that attempted a big bang. Also, Mabert et al. discuss the differing ERP benefits experienced by large versus small companies, and the degree of customization in each. Examining several such topics places the student results into perspective.

3.3 Step 3: Hands-on ERP Exercise

The final one-week component of the teaching approach is to provide some exposure to ERP software. As an SAP University Alliance member, John Carroll University has access to both SAP and SAP's practice database for a company called IDES. The students are given hands-on experience in three areas. First they process an order through to invoicing, so that they experience a business process. Second, they locate answers to business questions by searching the IDES database, so they see the importance of centralized data. And third, they explore some of the "drilldown" capabilities of the Logistics Information System, emphasizing the decision support features of SAP. Universities with other ERP systems can provide similar exercises.

Universities without ERP systems can experience the functionality of an ERP application through several good CBT courses. In particular, NetG offers tutorials which guide the student through SAP screens at their own pace, without the need for a live system. This is sufficient to provide them with an "ERP experience." In fact, John Carroll has very successfully used NetG tutorials even in advanced ERP courses that had access to SAP software. The tutorials allow students to quickly master the manual interface aspects of SAP along with many of the primary business processes contained in each SAP module. Classroom lectures then can focus entirely on more conceptual aspects. A good analogy can be made with respect to courses using Microsoft Excel. Students can master the manual interface outside of class using tutorial material, and classroom discussions can show how to use Excel to solve problems.

4. DISCUSSION

The above section has described three interactive exercises: an ERP case analysis, an ERP failure presentation, and an ERP hands on exercise. The goal of these exercises is to involve students in the fun and excitement of ERP systems, and to encourage them to pursue it further. In doing so, the exercises emphasize several points:

1) The ERP failure exercise develops appreciation for the SDLC before the students know what it is. In discussing the reasons their companies failed, students see that these reasons naturally fall into different categories – the SDLC steps. Thus they are exposed to the SDLC by creating it, giving contextual meaning to the knowledge.
2) Both the ERP failure exercise and the case study emphasize that success of a major IT project requires

tight coordination. Without management support, any IT project is doomed, and unresolved problems at one phase of an implementation can mushroom, creating bigger problems downstream. Again, this is learned through class presentations and discussions – not through lectures.
3) The ERP failure exercise emphasizes the enormous value of centralized data. As will be seen below, Hershey gained control over their pre-season inventory for the first time, and Nestle gained control of many of their vendors, also for the first time. Nestle felt that their SAP system saved them $325 million during the first two years of operation alone. Many of the failures turned into significant success stories and interesting business lessons, and the students will uncover the details during their Web searches.
4) The hands-on SAP exercise emphasizes that having centralized data at your fingertips can be exciting by giving a user enormous power to answer business questions. The exercise also shows a business process in action, allowing students to process an order and actually experience a flow of information.

The educational value of this exercise reaches beyond technology and project management issues to interject valuable business lessons that are placed in context. In particular, the class presentations bring out numerous interesting stories, which are valuable business lessons in disguise; and the students gain understanding of the business issues through self discovery. These lessons include issues related to corporate resource limitations, business ethics, the implications of cash flow, perishable inventory and seasonal sales, centralized purchasing and supplier contracts, and the list goes on. Furthermore, these interesting reports and analyses can be easily discovered by the students on the Web.

For example, in implementing its SAP system, Nestle discovered that it had 29 vendors of the ingredient, vanilla, and was paying a different price to each. This was due to a very confusing product numbering system. Another article discusses Hershey's failure. It reports that Hershey had built seasonal inventories in anticipation of Halloween demands, and much of the inventory was at off-site locations. Not all this information had been entered into SAP when they went live, and as a result the inventory could not be located when needed. Consequently, Hershey lost most of its Halloween sales. As a third example, Fox Meyer Pharmaceuticals had ambitious plans to install SAP and three bolt-ons simultaneously. While the project was underway the company accepted a new and unexpected $1 billion sale, and shortened the go live date to accommodate it. The resulting cold start failed (predictably) and the company went bankrupt as a result of the problems. As a fourth example, when Cleveland State University went live with a beta version, they had great difficulty registering students, preserving confidentiality of records, and paying faculty and staff.

Finally, the following lesson was learned by the instructor: students appreciate ERP best through self-discovery and interactive learning. Lecturing about the stages of the SDLC and ERP implementations is unexciting to students who have no experience and no frame of reference. By reading a novel and investigating a failure, they gain some degree of background, which they then can parlay into conceptual knowledge through discussions with others. They become involved in a learning process that is appropriately called "Welcome to ERP."

5. CONCLUSION AND FUTURE RESEARCH

The "Welcome to ERP" approach is a highly interactive learning process that draws students into the excitement of ERP systems. By working in groups and reinforcing each other's learning, students actually develop the stages of the SDLC – a far more meaningful experience than reading about them. In addition, the students learn the main implementation issues for major IS projects by studying and comparing some of the high profile failures. This gives them an appreciation of the immense amount of coordination involved for a project to succeed, and the numerous things that can go wrong – again, by doing it themselves, rather than being lectured about it. At the same time, with the emphasis on ERP, students get an early introduction to enterprise software and its importance to the business world of today.

The primary focus of the teaching approach – an exercise in analyzing ERP failures – can be incorporated into several different IS and management classes. Now that it has become a successful component of an elective ERP class for undergraduate business majors, the exercise will be added to a business IS core course taken mostly by junior-level students. More specifically, the exercise will replace the existing SDLC and ERP components that are currently covered in the IS class. Likewise, it also may be added to a required IS course in the MBA program.

In addition to its implications as a reproducible teaching exercise, the advancement of an SDLC-based analysis of ERP failures also has implications for IS researchers. Not only does the exercise add to the body of ERP pedagogical approaches, but it also offers a basis for positioning introductory ERP concepts within the foundations of the IS discipline. Such additions to ERP pedagogy in turn provide a basis for comparing, contrasting, and studying various approaches so as to advance and position the role of ERP within IS curricula. Furthermore, the exercise employs a learning process of constructing knowledge within a meaningful context thereby taking a constructivist route to teaching IS concepts (e.g., Chen, 2003). As such, the exercise adds to the body of constructivist pedagogical approaches and thus can be compared, contrasted, and studied with respect to the effectiveness of a constructivist process as the underlying learning technique.

At a more immediate level, the inclusion of the exercise in John Carroll's business IS class also will allow for some analysis of the teaching approach. First, as the exercise replaces current ERP and SDLC components in the IS course, it will allow for an evaluation of the effectiveness of the approach itself. Second, the incorporation of the exercise across courses will allow for a comparison of the fit and usefulness of the approach based on the focus and make-up of the class. In addition, the sequence of the components of the approach can be analyzed. For example, in an ERP class, is the assignment better employed at the end of the semester after a familiarity of the software (and its capabilities) has been developed (i.e., effectively switching the components of the approach so the software exercise precedes the SDLC exercise); or is it better being used as an introduction to ERP whereby it precedes any software exercises (as it is currently implemented)?

The importance of the SDLC-based teaching approach lies not only in the effectiveness of the exercise but also in the appeal and interest of the topic as it is presented to the students. It would not be surprising to learn that many business students find IS topics such as the SDLC and IS planning – topics that are covered in almost all IS texts – to be less than exciting due in part to their matter of fact coverage in the texts. Unfortunately, a lack of significance in presentation could translate into a lack of appreciation of the relevance of the topic as well as a lack of interest in the discipline. Ultimately, tradition must be balanced with innovation in efforts to develop new and effective teaching approaches that can position these topics as building blocks of curricular change.

6. REFERENCES

Ahituv, Niv, Seev Neumann and Moshe Zviran (2002), "A System Development Methodology for ERP Systems." Journal of Computer Information Systems, Spring 2002, Vol. 42, No. 3, pp. 56-67.

Al-Mashari, Majed, Abdulla Al-Mudimigh and Mohamed Zairi (2003), "Enterprise resource planning: A taxonomy of critical factors." European Journal of Operational Research, 2003, Vol. 146, No. 2, pp. 352-364.

Avison, David B. and Guy Fitzgerald (2003), "Where Now for Development Methodologies?" Communications of the ACM, January 2003, Vol. 46, No. 1, pp. 79-82.

Blackwell (1997). "Information System Methodologies." Blackwell Encyclopedic Dictionary of Management Information Systems, 1997, pp. 117-118.

_____ "Prototyping." Blackwell Encyclopedic Dictionary of Management Information Systems, 1997, pp. 186-187.

Bradford, Marianne, B. S. Vijayaraman and Akhilesh Chandra (2003), "The Status of ERP Integration in Business School Curricula: Results of a Survey of Business Schools." Communications of AIS, September 2003, Vol. 2003, No. 12, pp. 437-457.

Chen, Catherine (2003), "A Constructivist Approach to Teaching: Implications in Teaching Computer Networking," Information Technology, Learning, and Performance Journal, Fall 2003, Vol. 21, No. 2, pp. 17-27.

Davenport, Thomas H. (1998), "Putting the Enterprise into the Enterprise System." Harvard Business Review, July/August 1998, pp. 121-131.

David, Julie Smith, Harriet Maccracken and Philip M. J. Reckers (2003), "Integrating Technology and Business Process Analysis into Introductory Accounting Courses." Issues in Accounting Education, November 2003, Vol. 18, No. 4, pp. 417-425

Fincham, Robin (2002), "Narratives of Success and Failure in Systems Development." British Journal of Management, March 2002, Vol. 13, No. 1, pp. 1-14.

Fox, Pimm (2003). "The Art of ERP Done Right." Computerworld, 5/19/2003, Vol. 37, No. 20, pp. 22-23.

Henderson-Sellers, Brian and Julian M. Edwards (1990), "The Object-Oriented Systems Life Cycle." Communications of the ACM, September 1990, Vol. 33, No. 9, pp. 142-159.

Iivari, Juhani, Rudy Hirschheim and Heinz K. Klein (2000), "A Dynamic Framework for Classifying Information Systems Development Methodologies and Approaches." Journal of Management Information Systems, Winter 2000/2001, Vol. 17, No. 3, pp. 179-218.

Jacobs, F. Robert and D. Clay Whybark (2000), Why ERP: A Primer on Sap Implementation, Irwin McGraw-Hill, 2000.

Jayaratna, Nimal (1994), Understanding and Evaluating Methodologies, NISAD: A Systematic Framework. Maidenhead, UK: McGraw-Hill, 1994.

Landry, Jeffrey P., J. Harold Pardue, Herbert E. Longenecker Jr., and David F. Feinstein (2003), "A Common Theme for IT Degree Programs." Communications of the ACM, November 2003, Vol. 46, No. 11, pp. 117-120.

Langenwalter, G. (2000), Enterprise Resources Planning and Beyond: Integrating Your Entire Organization. St. Lucie Press, Boca Raton, FL, 2000.

Mabert, Vincent A., Ashok Soni and M. A. Venkataramanan (2003), "Enterprise resource planning: Managing the implementation process." European Journal of Operational Research, 2003, Vol. 146, No. 2, pp. 302-314.

Management Outline (1998), "Systems development life cycle." British Journal of Administrative Management, May/June 1998, Issue 9, p. 15.

Markus, M. Lynne, Sheryl Axline, David Petrie, and Sheryl Cornelis Tanis (2000), "Learning from adopters' experiences with ERP: problems encountered and success achieved." Journal of Information Technology, December 2000, Vol. 15, No. 4, pp. 245-265.

Parr, Anne and Graeme Shanks (2000), "A model of ERP project implementation." Journal of Information Technology, December 2000, Vol. 15, No. 4, pp. 289-303.

Plyler, Robert W. and Kim Young-Gul (1993), "Methodology myths." Information Systems Management, Spring 1993, Vol. 10, No. 2, pp. 39-44.

Ramiller, Neil C. (2003), "Making the Case: The Systems Project Case Study as Storytelling." Journal of Information Systems Education, Summer 2003, Vol. 14, No. 2, pp. 153-166.

Schambach, Thomas P. and Kent A. Walstrom (2002), "Systems Development Practices: Circa 2001." Journal of Computer Information Systems, Winter 2002-2003, Vol. 43, No. 2, pp. 87-92.

Stewart, Glenn, Guy Gable, R. Andrews, M. Rosemann, and T. Chan (1999), "Lessons from the field: A reflection on teaching SAP R/3 and ERP implementation issues." Americas Conference on Information Systems, Milwaukee Wisconsin, August 16-19, 1999.

Umble, Elisabeth J., Ronald R. Haft, and M. Michael Umble (2003), "Enterprise resource planning: Implementation procedures and critical success factors." European Journal of Operational Research, 2003, Vol. 146, No. 2, pp. 241-257.

Vickers, Margaret H. (1999), "Information technology development methodologies." Journal of Management Development, 1999, Vol. 18, No. 3, pp. 255-272.

Watson, Edward and Helmut Schneider (1998), "Integrating the SAP R/3 System into an MIS Program, Americas Conference on Information Systems, Indianapolis, August 16-19, 1998.

AUTHOR BIOGRAPHIES

Richard Grenci is an Assistant Professor of Management at John Carroll University in Cleveland, Ohio. Rick holds a Ph.D. in Management Information Systems from the University of Texas at Austin. His research interests include electronic commerce, online customization, and pedagogical issues in IS. He has published and forthcoming articles in Communications of the ACM, Computer Supported Cooperative Work Series, and the Journal of Computer Information Systems.

Bradley Hull, Assistant Professor of Management at John Carroll University in Cleveland, Ohio. Brad holds a Ph.D. in Operations Research from Case Western Reserve University, and a MS in Operations Research from Stanford University. He has taught courses in MIS, Operations Management, Logistics Management, and Enterprise Software at JCU since 1999. He is JCU's contact person for the SAP University Alliance. Previously, he held supply chain management positions at British Petroleum.