Re-engineering the Software Development Process for e Learning Development

Lawrence McIntyre
IT Project Manager eHR
GE Power Systems “InsideGEPS”

Abstract

The purpose of this paper is to present a logical argument that will defend the position that regardless of organizational dynamics within an organization's IT department, process change is needed in the development of e Learning if the organization hopes to be able to quickly and economically develop and delivery effective web-based training. The argument presents the "as is" process that the organization is using for all application development and offers an alternative process for the development of e Learning. The argument is framed in an understanding of the principles of organizational behavior as they manifest themselves in an IT organization.

"The very basics of management thinking such as planning, organization, control, development of people, culture and climate, and leadership change as circumstances change. Lessons learned at one stage in a career must be restudied within each and every business framework encountered." (Edwards 2001) Arrogance, ego, and inflexibility to new ideas and new processes will cause problems in every management area.

In most successful businesses, the IT department is a moving force in almost every company initiative. It is involved in every other department because of the support it provides in the design, development, or selection of digital tools that each department needs for quick cycle time of its assigned tasks.

The Human Resource department needs tools that guide the Human Resource Managers (HRMs) through the staffing and hiring process. These tools provide automated workflow processes that aid HRMs in making appropriate decisions by using algorithms that ensure EOC compliance. In the end, they (the tools or the HRMs?) help the company by mitigating legal risks.

The Sales force relies on the IT department to design and develop web applications that the salespeople can access from any global position. The need for accurate and fresh product information is paramount to the sales force's success in this global marketplace.

The Legal group needs the talents of the department to design and develop a document management system that allows multiple users to access and edit the document. This requires a sophisticated version control application to ensure legal contracts and addendums can be accessed, edited, and shared throughout the company.

The Business Development department needs a robust application that can control the hundreds of acquisitions that are considered or completed throughout the year. This application needs to be sophisticated enough to accept known values and calculate short-term, long-term, and life cycle cost benefits of each acquisition.

Each department's need is crucial to its members. The need to develop an application as quickly as possible is essential. The software development process needs to be continuously evaluated.

The analysis of the process must produce information and knowledge that can be shared and can be used to quickly re-engineer the workflow and processes. Morgan (1997) tells us that when a continuous process of information exchange is created between a system and its environment the system can monitor and self regulate and thus act in an intelligent manner. An intelligent system is needed in today's IT departments.

The purpose of this paper is to analyze an existing development process and suggest changes that will serve as the basis for improving the process to accommodate the quick and economical design, development, and deployment of Web-based e Learning programs.

Present Application Development Process

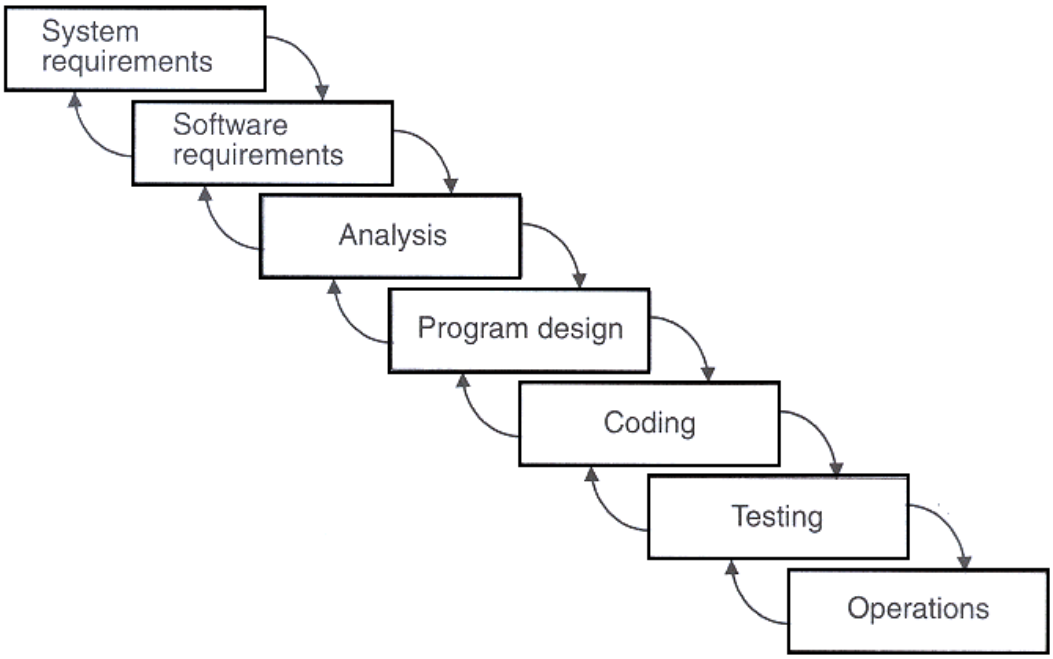
Application development is based on a workflow. The workflow describes the process as well as the responsibilities of the individual players in that process. If and when the process breaks down, it needs to have some level of change applied (to rectify the break down or make improvements?). A familiar metaphor that is used to describe workflow is the "hydraulic or plumbing metaphor." (Poyssick 1996) The workflows of most processes resemble a plumbing system, generally a tangled system of pipes pieced together of the years: a mixture of new and old technology coexisting in the same environment. Imagine a house built 50 years ago. The plumbing system contains iron, copper, and PVC pipes and combines old and new fixtures that are replaced only when leaks develop or a room is remodeled. Although this metaphor doesn't lend itself to well to the technology of today's IT departments it accurately describes the mentality of those in the department. All project managers, lead architects, technical leads, and programmers do not think alike. They all come from different places and different experiences. Some are older, as are their ideas of process, and some are much younger and thus embody new ideas about process. Regardless, the metaphor suggests that there will be clogs and closed valves as well as leaks throughout the system and there is a need to address these problems and make changes where appropriate. The vitality and function of the system depends on it.

When approaching the task of process re-engineering, it is important to realize the barriers that are placed in front of the perceived progress. Morgan points out that organizational dynamics play an important role in any change that takes place. The idea of studying business metaphors is rooted in the belief that they provide a new way of looking at and evaluating business organizations and their

internal dynamics. He also points out that metaphors can distort the reality and therefore no single metaphor will ever give us a completely clear point of view. Instead we need to become skilled at finding fresh ways of seeing, understanding, and shaping decisions in different situations. (Morgan 1997)

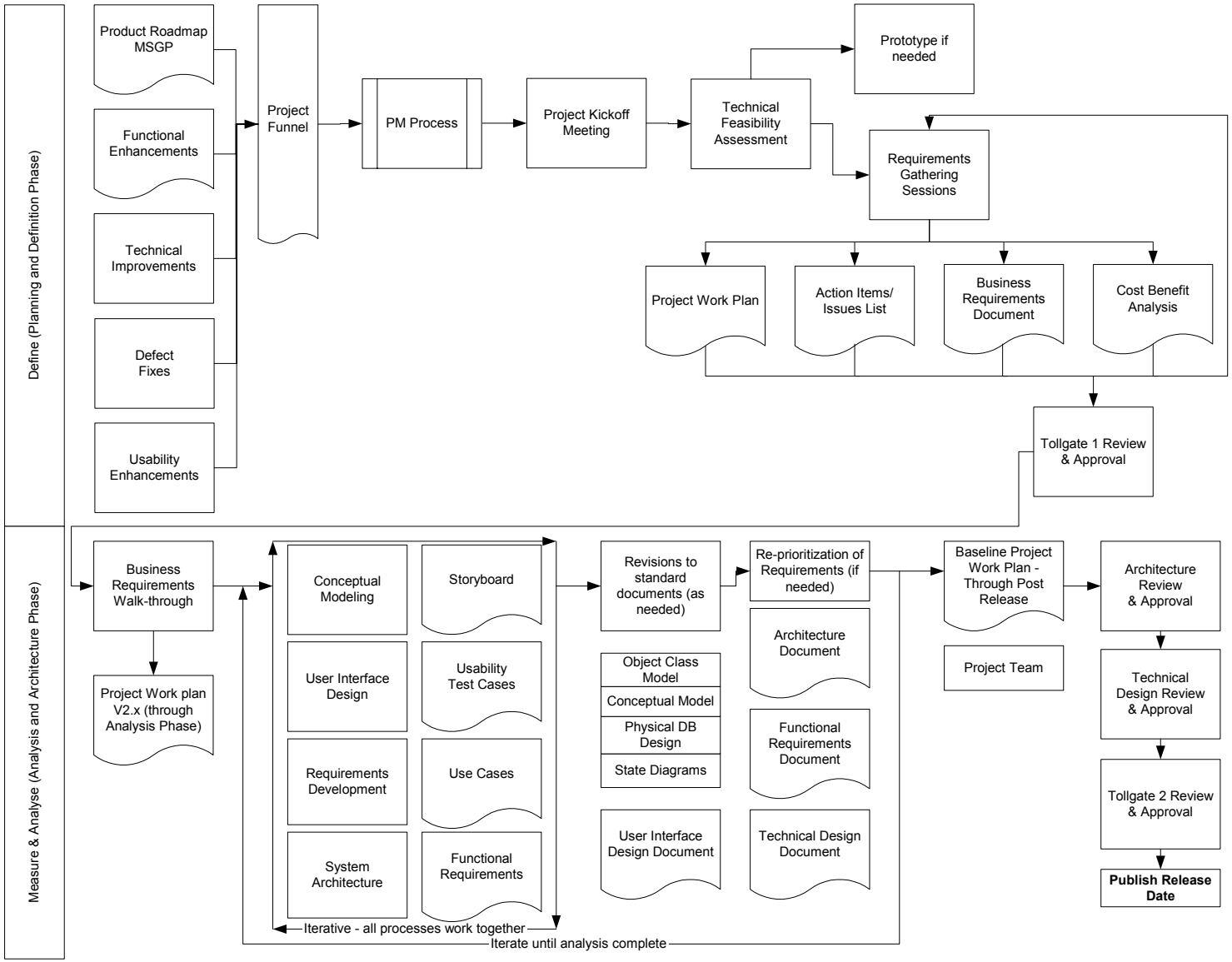
Software development is highly unpredictable. (Walker 1998) Only 10% of software projects are delivered successfully within the initial budget and schedule estimates. The management methods used play a larger role in the success of the project than do the technology advancements. The amount of redesign, change in direction and recoding during the development is a sign of an immature or mismatched process. It is extremely important to adjust the process to the customer's needs and requirements.

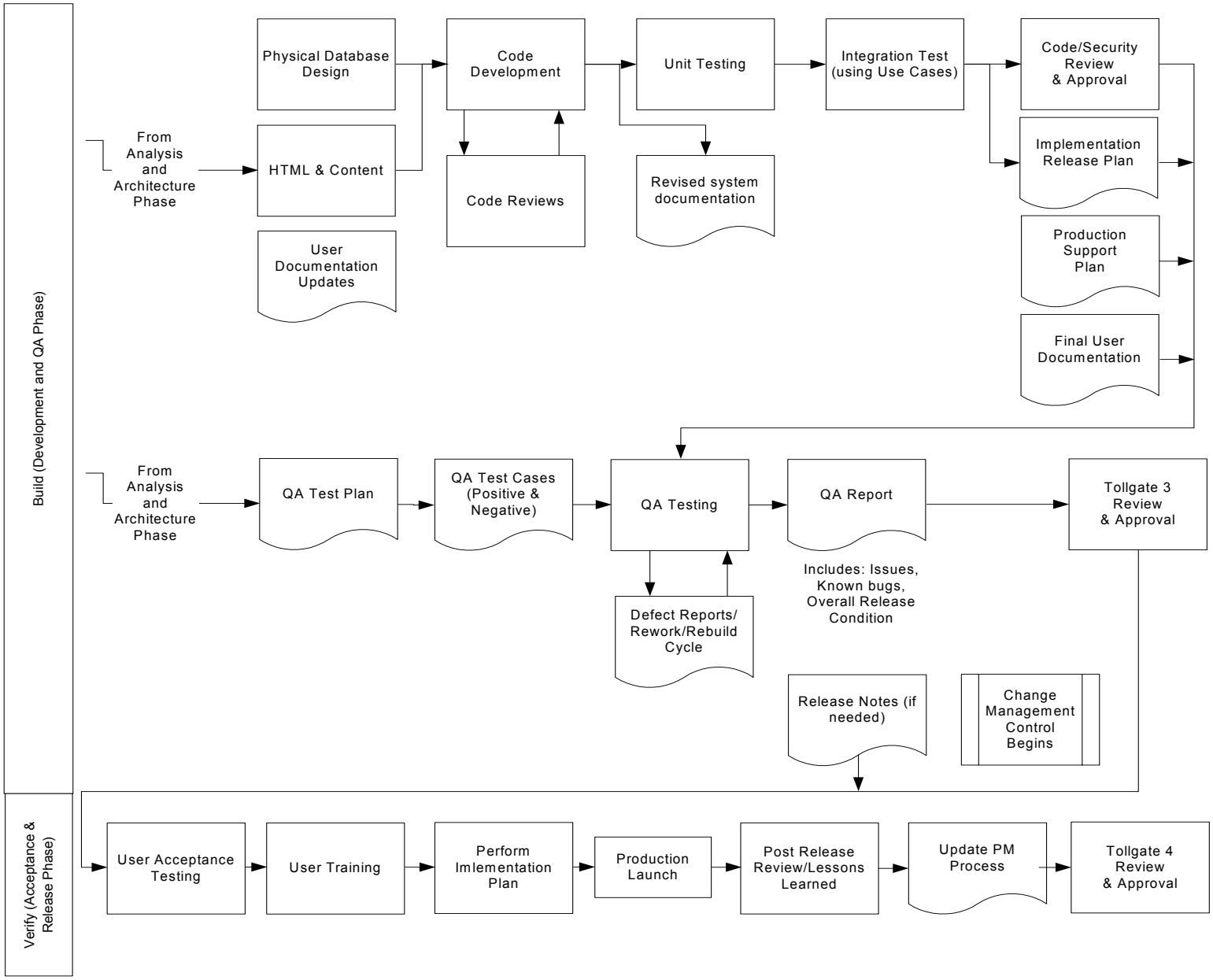
The process used by most development teams is known as the Waterfall Model. There are several derivations to this model but it is widely accepted as the "conventional" software process. There are two essential steps common to all development processes: analysis and coding. But in order to manage and control the development process, several more steps need to be added. These steps include system requirements definition, software requirements definition, program design, and testing. (Walker, 1970)



The above workflow describes the Waterfall Model for application development.

Most businesses add steps that add detail to each of the steps in the Waterfall Model. The following process flow charts show a more detailed look at the “as is” process and its phases. The more granular the process is, the higher the rate of success.


Define (Planning and Definition Phase)
Measure & Analyse (Analysis and Architecture Phase)
Product Roadmap MSGP
Functional Enhancements
Technical Improvements
Defect Fixes
Usability Enhancements
Project Funnel
PM Process
Project Kickoff Meeting
Technical Feasibility Assessment
Prototype if needed
Requirements Gathering Sessions
Project Work Plan
Action Items/ Issues List
Business Requirements Document
Cost Benefit Analysis
Tollgate 1 Review & Approval
Business Requirements Walk-through
Project Work plan V2.x (through Analysis Phase)
Conceptual Modeling
Storyboard
User Interface Design
Usability Test Cases
Requirements Development
Use Cases
System Architecture
Functional Requirements
Iterative - all processes work together
Revisions to standard documents (as needed)
Object Class Model
Conceptual Model
Physical DB Design
State Diagrams
User Interface Design Document
Re-prioritization of Requirements (if needed)
Architecture Document
Functional Requirements Document
Technical Design Document
Iterate until analysis complete
Baseline Project Work Plan - Through Post Release
Project Team
Architecture Review & Approval
Technical Design Review & Approval
Tollgate 2 Review & Approval
Publish Release Date


Build (Development and QA Phase)
Verify (Acceptance & Release Phase)
Physical Database Design
Code Development
Unit Testing
Integration Test (using Use Cases)
Code/Security Review & Approval
From Analysis and Architecture Phase
HTML & Content
Code Reviews
Revised system documentation
Implementation Release Plan
Production Support Plan
User Documentation Updates
Final User Documentation
From Analysis and Architecture Phase
QA Test Plan
QA Test Cases (Positive & Negative)
QA Testing
QA Report
Tollgate 3 Review & Approval
Includes: Issues, Known bugs, Overall Release Condition
Defect Reports/ Rework/Rebuild Cycle
Release Notes (if needed)
Change Management Control Begins
User Acceptance Testing
User Training
Perform Imlementation Plan
Production Launch
Post Release Review/Lessons Learned
Update PM Process
Tollgate 4 Review & Approval

Analysis Phase

The process actually starts long before the IT team begins their work. In the "political metaphor" discussion, Morgan (Morgan 1997) tells us that there is a great deal of "wheeling and dealing" going on in business today in which different people attempt to advance specific interests. In most businesses, any need for an application leads to the need for a business case. As part of the business case the project champion most likely participates in interdepartmental negotiations. The cost benefit of such an application is always stronger when the application can be shared among two or more departments. So the application's champion attempts to build liaisons to increase the odds for company buy-in. Most companies have some type of "tollgate" system in which financial gatekeepers make a go/no-go decision on the project based on the cost benefit analysis. Once approved, the application's champion can solicit the IT team's services.

The Business Requirements phase is an exercise in political correctness. "An organization's politics is most clearly manifest in the conflicts and power plays that sometimes occupy center stage and in the countless interpersonal intrigues that provide diversions in the flow of organizational activity." (Morgan 1997) A business analyst interviews the functional team to gather requirements. The political structure in the organization is immediately apparent and it needs to be leveraged to ensure that the requirements reflect the actual functional needs. More often than not, there's considerable disagreement on what the "as-is" or current process is and what the "to-be" or future process needs to be. So the IT team works to discern which functional team member has the "political power" in the group and maintains an alliance with that individual. This phase produces an important deliverable that is sometimes called the Business Requirements Document (BRD) which defines in detail the business requirements for the application. It is the initial design document and requires the customer's sign-off.

During the analysis phase, a Systems Analyst (SA), who uses the BRD, applies technical experience and designs an IT solution. The SA designs the solution based on the organization's technology stack (define technology stack), infrastructure, and on the industry's best practices. The organization has developed software specifications based on experience, technology analysis, and the desire to improve performance. These specifications serve as a guideline when designing a new application because the organization has applied what it has learned to its process. Bateson (Argyris

1972) refers to this applying of knowledge as deutero-learning (second order learning). The focus of this example is on single-looping learning because the emphasis is on learning for effectiveness. This phase produces a deliverable that is called a Functional Specification, which outlines the functionality of the application, the use case models, and in some cases a mock-up of all of the screens.

Design Phase

The next stage is sometimes call the program design phase. It usually has two very important deliverables. For the first deliverable, the Functional Specification is evaluated and the Tech Lead creates the Architecture Design, which lays out all of the interactions that take place when the user accesses the application. It details the interconnections that exist between the front-end of the application and databases or sub-routines on the back-end. The second deliverable is the final deliverable in the design effort. The Technical Specification Document details the technology that is used in the development of the application. It is the "assembly instructions" for the development team and it needs to be both accurate and efficient. The design is sent to the Chief Architect who reviews the Functional Specification, User Interface Specification, and Architecture documents to ensure consistency, appropriateness, and good robust design. These reviews also assist the business in aligning functional requirements with other projects and provide guidance and advice to project teams on potential technical solutions, directions, and recommended vendor products. In essence, the Chief Architect is a gatekeeper. The position brings with it a considerable amount of responsibility and power. It is often the source for clogs in the plumbing and cause for lost time in the development cycle. This individual is seldom concerned with the customer's desires and therefore operates under a completely different culture as the rest of the business. Conversely, the IT team is customer centric by nature. It works hard to deliver a solution to the functional teams and is forced to play the role of plumber, snaking the pipes of the system burdened by the metaphorical "old" pipes.

Development or "Build" Phase

After the sign-off on the architecture, the application is developed. The development team demonstrates the application to the functional team members and they are allowed to perform a User Acceptance test cycle. The defects are reconciled and the code is reviewed for compliance with the infrastructure standards. The application is then sent to a Quality Assurance (QA) testing facility for complete testing. It is installed on a mirror infrastructure and tested to ensure that it does not

influence other applications. QA can be another clog in the plumbing. There can be several issues that cause delays in the testing process including version control issues, incorrect data images, backlogs, and poorly trained testers. In addition, like the Architecture and Technical Specification review body, the QA facility may operate under a different culture than the rest of the business. QA teams tend to have a "primadonna" attitude because the provided service is vital to the process.

Implementation Phase

After the application testing is completed, the gatekeepers grant launching permission and the application is published to the production environment. Postproduction activities such as feedback and version generation improve the application over time.

The process, as described above, is crucial to the successful design and deployment of applications on a shared infrastructure. Because a business has many departments soliciting the efforts of the IT team and those efforts will be published to the same infrastructure, its paramount to ensure that the applications are designed and built to the same specifications. In most cases, these applications process user-entered data. That data processing can be as simple as database field population and subsequent reporting, or as complicated as calculating algorithms that produce spreadsheets indicating the net value of the company. In addition, the security of this data must be protected from unwanted users.

The major problem with most companies' IT policies is that they force everything through the same model. Little consideration is given to the scale of the project, the number of use cases, the size of the team or the complexity of the project. Looking at the standard process flow, it is easy to see that additional process steps require additional people and therefore, additional cost. And yet, some companies insist that a simple JavaScript Page (JSP) Web page development project go through this same process. Many companies insist that even a short 12 week e Learning development project, such as one utilizing JSP, Hypertext Markup Language (HTML), and Flash, go through this same process.

Walker tells us "software development efforts span a broad range of domains. While there are some universal themes and techniques, it is always necessary to tailor the process to the specific needs of the project at hand." (Walker 1998) Some may say that forcing e Learning development through the same process as developing an application is like shoving a square peg into a round hole.

The e Learning Development Process

In the introduction of his book Knowledge of Organizations, Laurence Prusak says "the only thing that gives an organization a competitive edge is what it knows, how it uses what it knows, and how fast it can know something new." Speed is essential. Speed is the most important characteristic of the digital world. It is either a competitive asset or liability. (Prusak 2001) Every member of the business needs to think about knowledge, learning, training and information. It does not matter whether you are a manager, a vice president, the CIO, or the CEO; everyone must ask, "What is the role of learning in this organization?" "What part do I play in building an appropriate infrastructure?" "How do I justify learning and e Learning to the business?"

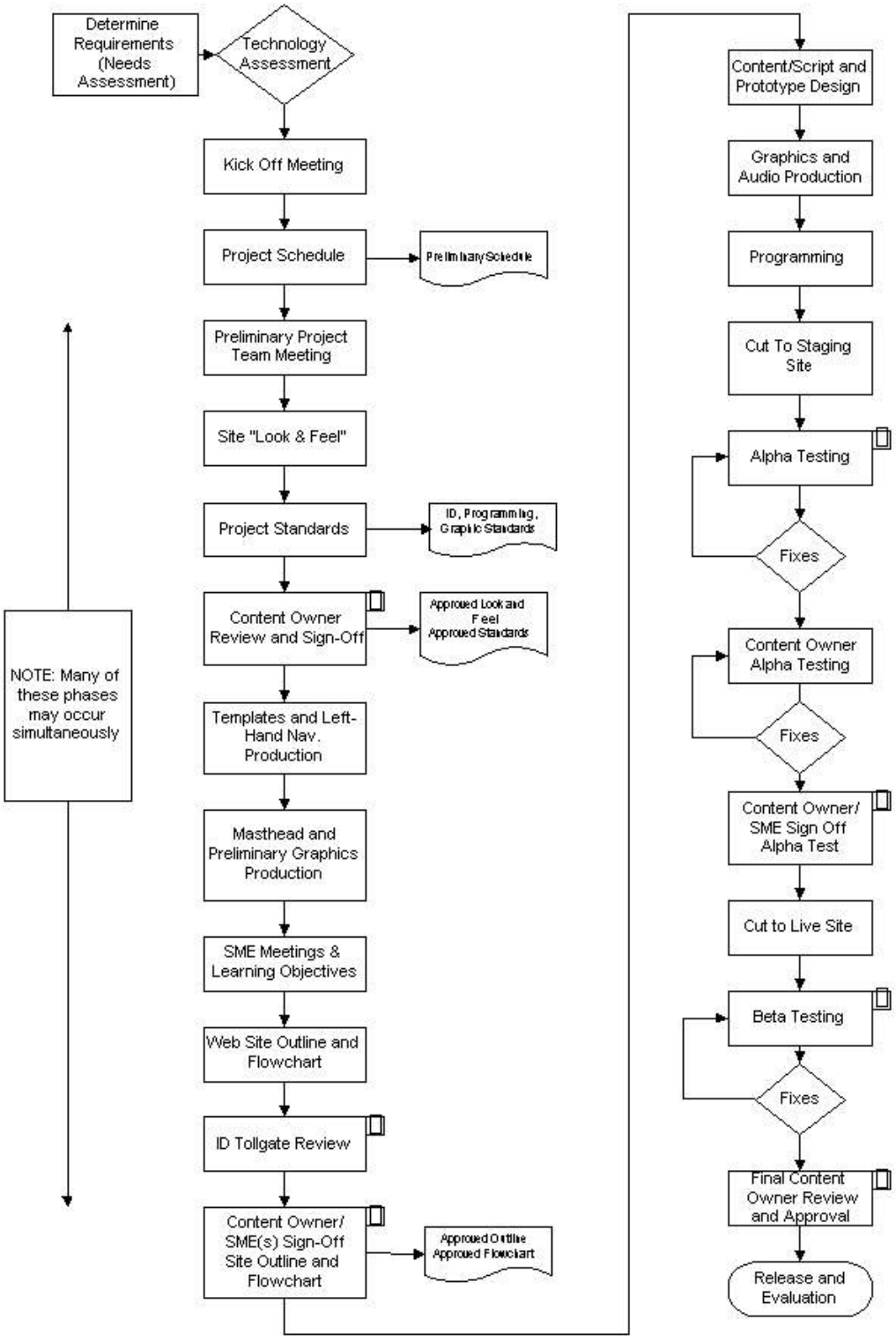
This first step in this "revolution of speed" is to accept a development process that is tailored specifically to e Learning. It is time to stop shoving a square peg into a round hole.

Phase	Analysis	Design	Development	Implementation	Evaluation
	Purpose: Identify the probable causes for a performance gap	*Purpose:* Verify the desired performances, the learning tasks, and the appropriate testing strategies	*Purpose:* Generate and validate the training materials	*Purpose:* Prepare the training environment and conduct the training	*Purpose:* Assess the quality of the instructional products and processes both before and after implementation
Procedure	➢ Conduct a performance assessment ➢ Determine goals ➢ Conduct a learner analysis ➢ Conduct a resource analysis ➢ Determine the likely delivery system	➢ Conduct a task inventory ➢ Compose objectives ➢ Generate testing strategies ➢ Calculate return on investment	➢ Generate instructional strategies ➢ Select or develop supporting media ➢ Develop the Learner Guide ➢ Develop the Facilitator Guide ➢ Revise using formative evaluation data ➢ Conduct a Pilot Test	➢ Select, prepare and schedule learners ➢ Select, prepare and schedule facilitators	➢ Determine quality assurance criteria ➢ Select evaluation tools ➢ Conduct evaluations
Product	Analysis Summary	Design Brief	Development Summary	Implementation Plan	Evaluation Plan

ADDIE Instructional Design Model

e Learning development teams utilize the Instructional Systems Design Process to design appropriate instruction. There are several models including the Carey, Dick, ADDIE and RAPID. Most teams design a process that is a customized version of these examples. The ADDIE Model is broken down into 5 steps, Analysis, Design, Development, Implementation and Evaluation and most teams follow a similar model.

e-Learning (Web-Delivery) Development Process



Customized Process Map based on ADDIE

Analysis Phase

The first step in the instructional design process is to meet with the client to determine if there is a need for training. (Alessi, Trollip 2001) A needs assessment may be conducted to identify the gaps between current behavior and desired or required behavior. An analysis of the problem and target audience makes it evident as to whether a training solution is recommended. If a training course is the answer to the client's current situation, a job and or task analysis may be conducted to determine performance requirements. The goal, target audience, and performance objectives of the course should be established before deciding on a delivery method.

The client, manager, and Instructional Designer (ID) determine which type of presentation method best answers the client's present situation. The advantages and disadvantages of training via video, CD-ROM, Internet and intranet sites, stand-up instruction, paper-based materials, electronic performance support systems, and other types of delivery methods are considered. The process that follows is based on the selection of the Web as the training delivery method.

Once the decision is made to produce a Web-based project, a kick-off meeting is held. The attendees include the client or client's representatives, manager, project ID, and programmer. The topics discussed in this meeting include the following:

- Goal of the Web site
- Target audience
- Needs assessment (if performed)
- Job/task analysis (if performed)
- Development process
- Release date
- Selection and role of Subject Matter Experts, or SMEs
- Resources required from client/SMEs
- Voice/presentation style
- Use and mix of various media
- Prototype (if applicable)
- Hardware/platform (network, print, security, database, browser)
- Connection method (LAN vs. Modem)
- Re-usability in future projects
- Help, glossary, and reference materials
- Evaluation

The project team is assembled and consists of the project ID (Instructional Designer), project Web programmer, project graphic artist, and Quality Assurance (QA/QC) representative. The project team meets to discuss the issues raised in the kick-off meeting, making sure that everything is understood and is on a to-do list. The meeting is also the time to discuss contingency plans for any problems that may arise. Additionally, the project team brainstorms the "look and feel" of the site. Project standards are also developed and modified.

Design Phase

The project ID organizes the objectives into a Website Outline (WO) and produces a flowchart that represents the Website. The flowchart allows the client, Subject Matter Experts (SME) and project team to review the site structure, objectives, interactions and strategies. The project ID also creates the first draft of the project schedule.

The project team creates a prototype of the screen design. The design should include navigational strategies, color schemes, specific layouts, and use of frames. When the screen design is complete, the project team reviews the prototype and provides feedback.

Once the Website Outline is completed, the project ID conducts a meeting to evaluate the Website Outline, flowchart, and project schedule. The project team, the Instructional Designer and manager attend. In preparation for the tollgate review, the project ID creates a PowerPoint presentation that demonstrates the flow and feel of the course. The tollgate review includes a sign-off.

At this point, the client and /SMEs review and approve the project standards and the look and feel of the screens. The client and SMEs also review and approve the Website Outline, Website flowchart, and project schedule. This phase also requires a sign-off.

Development Phase

The SMEs and project ID develop the content for each of the Website pages. This task includes gathering the reference materials required and the resources necessary to develop the content and media design.

The project ID creates the first draft of the project prototype (Preliminary Design Prototype). The project ID also creates the graphic sheets as part of the Design Prototype. Consultation with the SMEs may be required throughout the development of the Design Prototype.

The project ID creates the graphic sheets for the Mastheads (or headers) and Left-hand Navigation for all the pages that require them. Production begins using these sheets.

A technical writer reviews the Preliminary Design Prototype to check for grammar, style, spelling, punctuation, and discusses any editorial suggestions with the project ID. The project ID revises the Preliminary Design Prototype as necessary.

The Preliminary Design Prototype is posted on a testing server. At this stage, only the graphics for the masthead (header) and Left-hand Navigation have been created. The client and SMEs review the Preliminary Design Prototype and provide the project ID with comments and suggestions for revision.

The project ID revises the Design Prototype as necessary, based on the recommendations of the client and SMEs review. Additional resources and dialogue may be required from the client and SMEs, depending on the nature of the revisions. There is a sign-off for this phase. Final production may not proceed until the client and SMEs have signified approval of the Design Prototype.

The project ID and project graphic coordinator review the site for completeness and conformance to project standards. The site is then delivered to the production assistants for alpha

testing. Any bugs (discrepancies) and suggestions are recorded in the bug-tracking database. The production assistant must sign off on the completion of all alpha testing.

Each department is responsible for fixing assigned bugs. As the bugs and suggestions are addressed, the department responsible for the fix changes the status either to send it to another department (if necessary), or to retest. After all the bugs and suggestions are waiting for retest, the production assistant will retest and change the status as appropriate. After this, another graphic artist and ID will test the site. The graphic artist is concerned with the graphics standards; the ID makes sure that the ID standards and site objectives are satisfied.

The client and SMEs are given access to the Beta version of the site for review and testing. If bugs are found, they are fixed in a manner similar to the process described above.

Implementation Phase

The client and SMEs perform a final review of the site and signifies, via sign-off, that it is ready for general release. The programmer replicates the site on the client server. The production assistant performs the final test of the site on the client Web server to ensure that no new bugs have been introduced. This task also requires sign-off.

Evaluation Phase

The site is released to the target audience as per client requirements. Evaluation of the site may be conducted by field trial, by gathering and analyzing user feedback, and by gathering and analyzing other information.

Conclusion

Davenport & Short define *business process* as "a set of logically related tasks performed to achieve a defined business outcome." A process is "a structured, measured set of activities designed to produce a specified output for a particular customer or market. It implies a strong emphasis on how work is done within an organization" (Davenport 1993). Both of the processes discussed in this paper meet Davenport's definition. But, they are different. One is a tool, which sets policies for the development of applications while creating a bureaucracy that impedes the speed by which applications can be developed and deployed. The other is a tool, which sets policies for the rapid

development of e Learning applications while maintaining control of the number of actors in the process and thereby reducing development cycle time and cost.

The learning organization metaphor that Morgan discusses, (Morgan 1997) shows that an organization must be able to accept change and learn from experience. In today's marketplace, most businesses want to be known for their e Learning strategy. The management understands the competitive value of their "human capital." Preparing a business for a shift to e Learning requires that it build a new learning culture. The first step to building that culture is to take down the barriers that make leveraging e Learning difficult. (Rosenberg, 2001) This paper presented two development models and proposes that the difference be recognized. The first model serves well the development of large software applications and large infrastructure implementations. It involves such a large bureaucracy, that the cost of supporting the process not only adds to the financial costs but more importantly, it adds to the duration of the cycle time. The average time required to successfully launch an application utilizing this model is 40 – 52 weeks. The second model is presented as a better solution for the development of e learning, which requires development to take place at "internet speed." This model calls for a smaller team, more dedicated to meeting the milestones in the schedule and more focused on the design and development at hand. The average time required to successfully launch an e learning program utilizing this model is 10 to 12 weeks. It is important to stop forcing the square peg into the round hole.

References:

Argyris, C. (1978) Organizational Learning: A theory of Action Perspective. Reading MA: Addison Wesley.

Alessi, S., Trollip, S. (2001) Multimedia for Learning. (3rd Ed.) Boston, MA: Allyn & Bacon.

Davenport, T.H. & Short, J.E. (1990 Summer). "The New Industrial Engineering: Information Technology and Business Process Redesign," *Sloan Management Review*, pp. 11-27.

Edward T. Reilly (2001, November The Business Case for Lifelong Learning, Retrieved November 20, 2001, from http://www.amanet.org/editorial/lifelong_learn.htm.

Hall, B., (1997) Web-Based Training Cookbook. New York, NY: John Wiley & Sons.

Heinich, R., Molenda, M., Russell, J., Smaldino, S., (1996) Instructional Media and Technologies for Learning. Englewood, CA: Prentice-Hall.

Prusak, Laurence, (2001) Knowledge in Organizations, Boston, MA: Butterworth-Heinemann.

Rosenberg, Mark, (2001) e Learning, Strategies for Delivering Knowledge in the Digital Age, New York, NY: McGraw-Hill.

Royce, Walker E., (1997) Managing Successful Iterative Development Projects: A Seminar on Software Best Practices, (Version 2.3), Menlo Park, California: Rational Software Corporation.

Royce, Winston W., "Managing the Development of Large Software Systems," *Proceedings of IEEE WESCON* (August 1970), 1-9.

Reisman, S. (1996) Multimedia Computing. Harrisburg, PA: Idea Group Publishing.