Untrained Users Derail Your Caboose? Learn to Get on Track!

Melanie Bentley-Smith
Institutional Business Systems - Jet Propulsion Laboratory

Introduction

You've implemented Oracle, and low-and-behold, your users can't bear it. And Surprise! They're making costly errors, finding creative work-arounds, and being mavericks with your system. Why, you ask? Because user training wasn't on track. This paper discusses how to build a proactive, comprehensive training program to get your users back on track!

Executive Summary

Six thousand creative individuals can really test the metal of an integrated system. We at the Jet Propulsion Laboratory (JPL) learned this from experience. A comprehensive, proactive training program could have prevented a lot of the havoc produced by our inventive and resourceful users.

During implementation and into our sustaining period, a simplistic training model was assumed. Training was provided as a one-time event. Classes were designed in a one-size-fits-all approach, and were narrowly focused on the tools, with little or no coverage on process or system integration.

We soon learned this was a grave oversight. For example, because our users were not educated on our new business processes, or on system integration issues, they often looked for, and found, their own work-arounds. Many times, this wreaked havoc on the technical end, causing our implementation team to work an unwieldy number of extra hours on fixes. Our complete set of training lessons learned is shared in this paper.

We determined several training solutions based on our lessons learned. For example, end-user training must incorporate business process and system integration. Classes must be designed to specific user audiences, and must be process-based rather than tool-based. Classes must also be designed in a progressive approach, to build on users' varying skill levels. All of our training solutions are presented in this paper.

When our training solutions are packaged together, the result is a proactive and comprehensive training program, which brings many benefits. Specifically, it serves as preventative maintenance for your integrated system and processes, in that it prepares your users for the dangers that come with the territory. It also anticipates, and meets, the education and training needs of all users, from new hires to experienced users. It considers training as a core requirement for successful system deployment and maintenance. This proactive approach to training includes several elements, which are presented in this paper. The resources required to implement this program are also discussed.

Evaluating training effectiveness is critical to ensure continuous improvement. Without it, even the best training program will not continue to meet ever-changing user needs. This paper concludes with critical measures for training success.

Background on JPL

JPL is a non-profit federally funded research and development center (FFRDC) which is operated under contract by the California Institute of Technology (Caltech) for the National Aeronautics and Space Administration (NASA). JPL implemented Oracle 10.7 applications, including Financials, Acquisitions, and Human Resources, in September 1998. We currently have 45 business applications, including Oracle, custom-built and bolt-on applications. We have a user community of up to 6,000 personnel, including up to 1,500 functional users. We will be upgrading to Release 11*i* in mid-2001.

Background on Our Implementation Training Approach

During implementation and into our sustaining period, a simplistic training model was assumed. Training was provided as a one-time event. Classes were designed in a one-size-fits-all approach, and were too narrowly focused on the tools. For example, while the training provided content on the applications' functionality, it did not include coverage on the corresponding processes. Therefore, it lacked a context for how the applications would actually be used on the job. Likewise, while the training covered possible errors within each individual module, it did not provide insight into the consequences of making errors in a fully integrated system. It lacked a sense of responsibility involved in using an integrated system.

This resulted in some serous issues. For example, there was virtually no buy-in from the users during the training. In class, the users quickly realized the difficulty of using the applications to do their jobs without being educated on how the application fit into the bigger process picture. Therefore, they left the training un-convinced that they really had to use this new system. Another issue was that users were let loose into our production environment without sufficient practice time. They made data entry errors, and ran concurrent manager jobs incorrectly. Our more creative users took the initiative and found their own work-arounds, not understanding the huge number of hours that needed to be spent on fixes behind the scenes.

Another factor worth noting was that in many cases, the training sessions were our customers first interaction with the implementation project. As already discussed, there was a lot of dissatisfaction and frustration during the training sessions. The consequence was a general sense of dissatisfaction and frustration with the entire project, and with the Oracle applications. Our customers felt alienated by the project, which lead to skepticism. This cost the overall project in a big way, the price being a loss of credibility right up front.

This loss of credibility may sound trivial, but it did have costly consequences and has been a huge effort to re-build. The consequence that we are facing right now is that we have a strong constituency of users against the upgrade to Release 11*i*. Specifically, they are trying to influence senior management to postpone the upgrade because we have not resolved all of the current issues. Of course, this is not an option, as Oracle is de-supporting 10.7 next year.

A Breakdown was Declared

Due to these problems, a breakdown was declared across all areas of the implementation project, including training. In regards to training, this led us to gather and analyze our training lessons learned in an effort to help determine how to solve the training breakdown.

An Analysis of our Training Lessons Learned

This section presents the training lessons learned, discussing the encountered problem, the resulting impact, and the solutions that are incorporated into the progressive training approach presented later in this paper. These lessons learned range from the initial implementation up through the current sustaining period.

1. Problem:

   Planning for training resources during the initial implementation occurred too late in the implementation project.

   Result:

   - Training resources were not adequate to meet the scope of the project.
   - Post implementation experience showed that more training through stabilization could have diverted early system problems.

   Solution:

   - Plan and resource load the training effort up front, to ensure that training is sufficiently budgeted and forecasted.

2. Problem:

The functional teams did not consistently plan for the training requirements during the design and validation of the training. They did not sufficiently plan for the time and resources that training would need from them.

Result:

- The training materials did not have the integrity they should have.
- The training team did not have the opportunity to provide insight into potential user problems up front.

Solution:

- The functional teams must understand, agree, and buy-into the training approach up front. They must include sufficient time and resources in their own plans.
- The training team must partner with the functional teams early on in testing and providing demonstrations to stakeholders. This will engage stakeholders, and increase the training team's knowledge of the process and tools.
- The training team must also partner with the functional teams early on to facilitate the capturing of potential user issues.

3. Problem:

During the initial implementation, the organizations throughout the Laboratory did not consistently know what to expect from training, nor did they consistently provide the required information to the training team.

Result:

- Customers did not feel they were properly informed of what to expect from training.
- The training team did not have accurate student information from all organizations.

Solution:

- There must be an effective enterprise-wide communication and support infrastructure, with designated (willing and able) points-of-contact, to assist in the gathering and disseminating of training-related information. This creates an environment of ownership and enrollment not only for training preparation, but also for the overall system and training implementation.
- Training communication must be fully integrated into the training process.

4. Problem:

An on-going training team was not maintained for system training after the initial implementation.

Result:

- Training resources were not sufficient after the implementation to have an effective, holistic training program.
- Once processes were better defined, and system integration was more clear, we did not have the training resources to provide training in these critical areas.

Solution:

- Establish a permanent, core training team. This team must continue on throughout the lifecycle of the system.

5. Problem:

During the implementation, users had no exposure to the new process, or tool, prior to the training. They did not know what was in it for them. They did not even know whether their management was aware of the process changes impacting their jobs.

Result:

- Users became too focused on the process changes early in the training class, and could not re-focus on the training materials.
- Users did not buy-into the new process during training, and could not understand how the training related to them.
- This created an ineffective learning environment during class.

Solution:

- Process changes, particularly those that impact job functions, must be clearly communicated to all customers and their management prior to attending classes.
- Students must have the opportunity to be walked through the process, and to see the tool, prior to attending the hands-on class.

6. Problem:

Training classes have been designed as a "one-size fits all" class, where one class is delivered to all users of a given module. This is inconsistent with the proven adult learning principle that adults learn new material by connecting it to what they have done in the past.

Result:

- The course may not have been relevant to the students, or connected to what the students had done in the past.
- Students have become confused and frustrated early on in class because they do not understand where the class is taking them.
- Students may not have understood how the material related to their jobs, or how they would use the material back at the office.

Solution:

- Because proper context is key to effective learning, classes must be designed for targeted groups of users. If different groups of users will use a given application differently, then training must be designed to meet their varying needs. For example, because project resource administrators use the Projects module differently than line administrators, separate courses should be offered. This will provide the proper context for the material, and will help ensure an effective learning environment.

7. Problem:

Training has been delivered as a one-time event. This is inconsistent with the proven adult learning principle that behavior is not changed from attending just one class. Nor is all material absorbed and understood from one session alone.

Result:

- Users fall into bad habits because there is no follow up training.
- If students do not remember what they learned in class, they will find their own way of doing things.

Solution:

- Because repetition is key to effective learning, multiple opportunities must be provided for users to attend training sessions.

8. Problem:

Training classes have been designed to include all possible transactions, from beginner level to advanced. All necessary information is provided in one class. Students are not provided an opportunity to develop foundational skills prior to learning advanced transactions.

Result:

- This creates an environment of information overload, where students take back too much information to be able to effectively apply it to their jobs.

Solution:

- Establish a proactive, progressive training approach, where classes are designed to build on the students' varying level of knowledge and experience. The beginning point would be the new hire, with training classes designed to targeted user levels all the way through to experienced users.

9. Problem:

System and process integration have not been sufficiently included in the training. Training has focused on how to use the system, rather than how to do one's job using the system. The training has not been revisited since the initial implementation to include these pieces due to lack of resources.

Result:

- Students do not understand the impact that they have on the system or process when they deviate from the standard.
- Students may not fully understand how to do their job in the system, nor do they understand how their job fits into the bigger picture.
- Students have found their own system and process work-arounds that have negatively impacted other groups, because they do not understand why to follow the rules.

Solution:

- Include appropriate system and process integration pieces into training. This would require that our training process become more of a collaborative effort with functional and technical areas.
- Re-focus training to teach users how to do their jobs using the system, fully integrating process into the class material.
- When a deviant transaction with a high negative impact is discovered, provide special training to "nip it in the bud."

10. Problem:

During implementation, and with subsequent new products, training was provided too far in advance of application rollout.

Result:

- Students forgot what they learned by the time the application is rolled out.

Solution:

- Provide a practice instance so students can use the material right away, so they will remember the material when the system is in production.
- If a practice instance is not available, the training should be rolled out no earlier than one week prior to the students' ability to use the system.

11. Problem:

The template used for training materials was not well received by the instructional designers and technical writers. Additionally, from a user standpoint it was incomplete because it did not provide screenshots and other graphics. This is inconsistent with the proven adult learning principle that adults vary in the way they learn material. The training documents were only geared towards those who are text-oriented.

Result:

- There was inconsistency in the use of the training template among the instructional designers and technical writers.
- Students who are picture-oriented did not find the training material effective.

Solution:

- Select and implement a documentation tool, or template, that will be useable and acceptable, to all those involved in the system training documentation process.
- Insure that this tool lends itself to inclusion of screenshots and process flows.

A Recommended Training Program

In response to these lessons learned, the following training activities are being implemented. When they are packaged together, the result is a proactive and comprehensive training program, which anticipates and meets the education and training needs of all users, from new hires to experienced users. It also serves as preventative maintenance for your integrated system and processes, in that it prepares your users for the dangers that come with the territory, thereby helping to diminish user errors. As already discussed, it is critical that these programs be targeted towards specific user audiences.

- Business system orientation at the new-hire orientation. This will introduce the concept and responsibility of an integrated system to new employees up front, and will let them know what systems are available and how to obtain access and training. It will also provide an overview of our on-line timecard.

- Process and tool overview for new users. This will provide an opportunity for new users to see the new process and tool in action prior to the hands-on class, thereby facilitating the students to be prepared to focus on the training material during the hands-on class.

- Oracle navigation for new PC users. This will help prevent the classes from lagging due to a new PC user falling behind.

- Foundation classes for basic, every-day transactions (process and tool). This will focus the initial training sessions on the transactions that the users need to know right away, and will build foundational skills in the system and process.

- Advanced topic classes for complex transactions (process and tool). This will provide advanced training to handle complex situations when the user becomes more savvy. The concept here is to build foundational skills first, then bring users in for more complex training.

- Event-based classes for time specific transactions. This could include special training for year-end close, special training for mid-year budget exercises, etc. This training will coincide with the event, thereby providing just-in-time training.

- Problem-solving workshops for real work solutions. Here, we request users to e-mail us in advance with problems they want to bring to the workshop. We can then schedule in advance the appropriate subject matter experts to attend. This allows users to obtain answers to problems one-on-one in real time, from the people with the right answers.

- Presentations for fixes and high-impact transactions. This will provide an infrastructure to present information regarding any new bug-fixes, and to address any transactions that have been identified as causing negative impacts to the system.

- Periodic open instructor office hours. This provides students who struggle in class with an opportunity to practice more, and ask the instructor specific questions in a comfortable environment.

- Web-based training. This serves as an excellent refresher, or an introduction to a new topic.

Defining Training as a Lifecycle

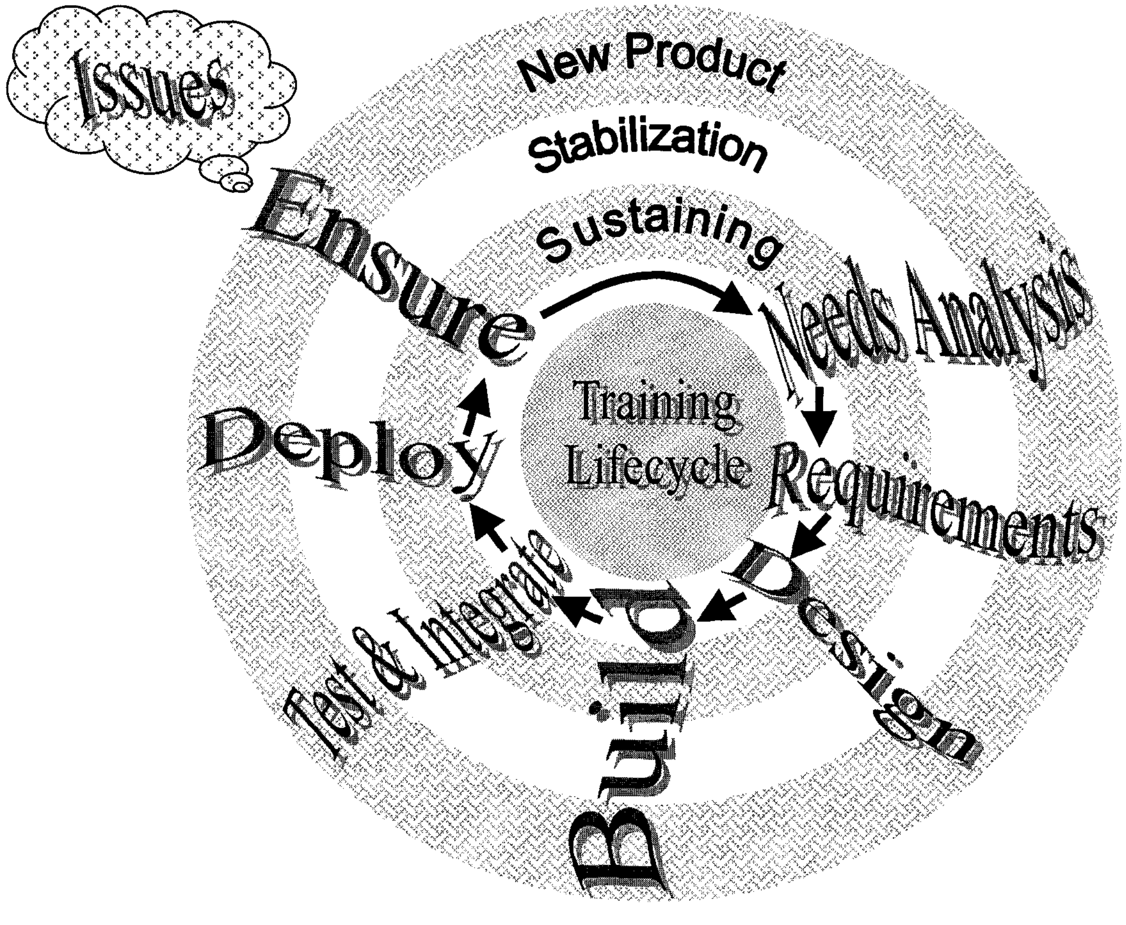
From the analysis of our lessons learned, it became clear that successful training is a core requirement for successful system deployment. We realized that we had to take a completely different approach to training, one that continued throughout the life of the system. Training must move with the system, and address changes as they occur. Our answer to this realization was to define training as a lifecycle. We defined seven steps to the lifecycle, all of which flow through the three states of our system: sustaining, new product, and stabilization. This section will first discuss the Training Lifecycle in the context of these three states, and will then define the seven steps of the lifecycle.

- **Sustaining State:** This refers to the training efforts associated with an on-going, stabilized business system. These efforts include on-going training delivery, training assessment, courseware enhancement and updates, and the planning and delivery of new, supplemental, or progressive courses in response to changes in user needs, or changes to the system or process.

- **New Product State:** This refers to the training efforts associated with the planning and delivery of new courseware in conjunction with the rollout of a new software product. This can range from the rollout of a new module to a major upgrade project. In comparison to the rollout of a new module, the scale of change increases substantially with a major upgrade project, which has the possibility to impose wholesale changes across all business applications.

- **Stabilization State:** This refers to the training efforts associated with the transition state between new product and sustaining, where the system is moving out of the new product state, and becoming the on-going, sustaining system. It is a separate state because a system undergoing stabilization may require changes to the software or business procedures, therefore making user needs more acute.

The following diagram illustrates the Training Lifecycle, and shows how the seven steps of the lifecycle extend throughout each training state.



This diagram illustrates that each state requires the same seven steps of the lifecycle. The arrows pointing from step-to-step illustrate that the process is continuous, thus indicating a continuous improvement model. The lifecycle steps are defined below. Attached to this paper is a sample work plan showing the activities that should fall under each step.

1. **Ensure:** The on-going assessment, or self-audit, of the effectiveness of training. Specifically, this is self-assessing the training product to ensure that it was effective, and satisfied its objectives. The effort includes course surveys and other tools/measures. When the assessment indicates there is a gap between the current training offering and needs, this leads to the need to conduct a needs analysis.

2. **Needs Analysis:** The assessment of the user community, and the determination of what they will need from training. This includes determining who needs training in which modules, and to what level of proficiency. This also includes assessing the users' current level of proficiency, to help determine the baseline. There is also an evaluation of the identified training gap to determine implementation priority and resources required.

3. **Requirements:** The determination and planning of what is required to fill the gap identified in the assessment. This could include establishing new training objectives, building new, supplemental, or progressive courses, broadening an audience, etc.

4. **Design:** The design of the training course or curriculum to meet the requirement determined above. This includes determining the appropriate format and presentation method, the content to be included, the appropriate audience, etc., in order for the class to meet its objectives.

5. **Build:** The development of course materials including student guides, quick reference guides, class exercises and activities, presentation materials, etc.

6. **Test & Integrate:** The validation of developed training materials. This includes but is not limited to piloting the training course/courseware.

7. **Deploy:** The scheduling, delivery and implementation of training within the user community.

The issues on the diagram refer to gaps between the training offering and the user needs. Issues can surface through changing user needs, the introduction of new functionality, new releases, system and process changes and major upgrades.

The lifecycle diagram shows that each of these steps are required in each training state previously defined. However, it should be noted that there is a varying magnitude and breadth of scope within each state. For example, in the sustaining state, a needs analysis must be conducted in order to define on-going training objectives, and to determine if the training is meeting the on-going needs of users. The new product state also requires a needs analysis. However, in this state, the needs analysis is larger in scope. This is particularly the case in a major upgrade, which typically has enterprise-wide impacts on training needs. The stabilization state also requires a needs analysis, but here, the analysis is done to determine if there are any system or process stabilization issues for which training must be provided.

Other parameters that vary in magnitude from state-to-state include: audience size; number of modules impacted; degree to which the processes are changed; degree of cross-functional impact on business procedures; level of new functionality; and impact on culture.

The resource requirements also vary among the training states. The new product state is the most resource intensive. This is true whether in conjunction with the release of one module, or with a major upgrade effort. However, a major upgrade effort requires substantially more resources than the release of an individual module. During the stabilization state, the resource demand begins to go down, until it reaches the baseline level required in the sustaining state. Training team resources are discussed in more detail later in this paper.

Defining training as a lifecycle provided another critical benefit. It served as the vehicle through which we were able to convince senior management of the need for on-going training support and resources. In other words, it served as the vehicle through which training was able to obtain buy-in from senior management. Prior to this time, we were unable to convince senior management of the critical need for an on-going system training team and infrastructure.

Implementing the Training Lifecycle

Through all of the information presented herein, we have realized that our training must be more user-driven and targeted to specific user audiences, based on the system and process functions they perform. In order to implement such a targeted training program, we must first understand our user community. To gain this understanding, we initiated a Laboratory-wide user needs assessment that will be completed in October 2000. Specifically, we are looking at each organization within the Laboratory to identify user categories and determine the tools and processes they use. We will then be looking at each user category, to determine the specific transactions they perform, and will be assessing their comfort level with the current tools and processes. We will also be gathering user recommendations for future process and tool enhancements.

This detailed information is necessary to enable us to develop training curriculum that is truly targeted to our different user groups, encompassing process and system integration, to best meet all of our users' needs. Providing this type of training now will help prepare our users for the Release 11*i* upgrade next year, as we do not want them to bring their old bad habits into the new system. It will also help dispel some of the resistance we are experiencing to this upgrade, as discussed earlier in this paper.

Measures for Success

Evaluating training effectiveness is critical to ensure a continuous improvement model, such as the Training Lifecycle presented in this paper. When evaluating training, there are four levels of effectiveness to consider: 1) Student satisfaction with the class; 2) the level of learning that occurred during and after the class; 3) the level of behavior that was changed as a result of the learning, i.e., the level to which the students applied the learned material to their job; and 4) organizational results, or return on investment, i.e., to what level did the organization achieve its goal as a result of the improved performance of the employees due to the training.

The following techniques may be used to measure the success of training:

- Course evaluations
- Follow-up questionnaires/assessments
- Student focus groups
- Partner with the help desk, and other support organizations
- Partner with key user groups
- Partner with system managers and functional subject matter experts

Training Team Composition

A core, on-going training team must be in place to adopt a continuous training model such as the one presented in this paper. The team would consist of a business system training manager, business system training specialists, a documentation administrator, and a training coordinator. In the event of a major upgrade, additional temporary contract resources may be required. The roles of these team members are outlined below.

- **Business System Training Manager:** The manager is responsible for setting the strategic direction of training, setting standards and making decisions for training, and managing the operations and staff. The manager is also responsible for training communications, and obtaining buy-in, at an institutional level. At the end of this section are tables that outline the specific manager responsibilities in the areas of planning, personnel, communication, customer support, design and deploy, and logistics.

- **Business System Training Specialists:** The training specialists conduct all aspects of the training function, including conducting needs assessments, identifying training objectives, designing curriculum and learning activities, developing course materials, delivering classes, and evaluating training effectiveness.

- **Documentation Administrator:** This person administers the documentation tools, defines controlled documents, enforces documentation standards, and writes desktop procedures using content collected from functional subject matter experts.

- **Training Coordinator:** The coordinator handles all scheduling, coordinating and administrative efforts for training.

Detailed Business System Training Manager Responsibilities:

Planning Responsibilities
Assess the scope of a new product, or a major upgrade.
Develop a training plan for new training offerings, including resource and budget requirements. This could be in conjunction with enhanced training, or training for a new product or major upgrade.
Establish training objectives.
Conduct a gap analysis between training program and the training objectives.
Determine training requirements to fill any identified gaps.

Personnel Responsibilities
Manage training personnel.
Select new team members as needed.
Establish temporary, contract training team for major upgrade.
Provide input for training personnel employee evaluations.
Assign instructional designers to functional areas for course design and development.
Direct and support the training team in designing and developing the courses.
Direct and support the training team in delivering the training.
Direct and support the training team in scheduling the training.

Communication Responsibilities
Form and maintain alliances with functional teams, key-end users, and management. These alliances will be key throughout the entire lifecycle process.
Design and maintain a Laboratory-wide infrastructure for training-related communication and support. This infrastructure will be key throughout the entire lifecycle process, to insure the training programs are adopted.
Design and implement a training communication strategy on the new product, or major upgrade, to address the concerns of users and management. This is also key to insure the training programs are adopted.
Facilitate the communication of training design and delivery plans to end-users on all training products/programs.
Facilitate user and management focus groups to capture and address training concerns.
Partner with functional organizations regarding any process training they offer, to ensure a coordinated training effort.

Customer Support/Troubleshooting Responsibilities
Anticipate possible issues with the user community.
Resolve issues with the user community.

Design and Deploy Responsibilities
Design and deploy a methodology to assess individual user proficiency level.
Design and deploy a methodology to measure training effectiveness.
Design and deploy instructional design and training delivery standards.
Facilitate the training team in forming alliances with functional teams.
Facilitate the training team in conducting a needs analysis.
Design and deploy a progressive, enhanced training program, based on identified requirements.

Logistics Responsibilities
Evaluate a documentation tool, or template, for use in training documents.
Facilitate the coordination and capturing of external courses attended by IBS and functional team members.
Address training facility needs.
Determine the requirements of the training instance, and facilitate the maintenance of the instance.
Work with Oracle, and IBS management and System Managers, to determine appropriate classes for members of IBS and the functional teams.

Final Thoughts

If there were three final thoughts that I could summarize for you to impress upon your company the importance of a proactive, comprehensive training program, they would be:

1. Effective and successful training is a core requirement for an effective and successful system deployment.

2. In order to be truly effective, training must be targeted to specific user groups based on the process and system transactions they perform. Training must also be process-based and address system integration.

3. In order to implement such a program, you must have senior management support and buy-in. Defining training as a lifecycle was the vehicle through which we obtained that support and buy-in.

About the Author

Melanie Bentley-Smith is currently serving as the System Training Manager for the New Business System (NBS) products. She initiated and defined the NBS Training Lifecycle in response to an analysis of the training lessons learned that she also initiated and led. She is currently responsible for all aspects of the training project plan for the Release 11*i* upgrade. She is also responsible for the day-to-day operations of the training effort and the training team, including a Laboratory-wide NBS user needs assessment. During the initial implementation, she designed and developed 25 courses for 11 Oracle and custom-built applications, and delivered 30 courses for 17 Oracle and custom-built applications to a user community of up to 6,000 employees. She has an MA in Human and

Organizational Development from Azusa-Pacific University, and a BA in Psychology from California State University, Long Beach.

The work described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology under a contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process or service does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

Appendix 1: Sample Work Plan for Implementing the Training Lifecycle for End-User Training

Needs Analysis
Assess the user community
Identify training audiences
Determine required skills and tools for each audience
Assess current proficiency level of each audience
Conduct gap analysis to determine training needs
Assess the scope of upgrade and impact on end users
Evaluate training documentation tools

Determine Requirements
Set end-user training objectives to meet identified needs
Develop an end-user training plan (including required resources) to meet the objectives
Define training communication needs
Determine training communication strategy
Decide on a training documentation tool that will meet the training objectives
Establish and refine instructional design and training delivery standards
Identify the mechanism required to schedule the training
Plan and implement the training schedule
Address training facility and equipment needs as required
Establish temporary, contract training team for new upgrade *(if upgrade scope requires additional resources)*

Design Training Courses and Materials
Design the curricula, including course outlines for each user audience identified in the needs assessment
Identify transactions to be included in each class, for each user audience identified in the needs assessment
Identify process pieces to serve as the context for the system transactions
Identify integration touch points for the transactions (system and process)
Based on the design of the courses, identify the training instance requirements
Communicate design plans to the Laboratory at large

Build Training Courses and Materials
Develop the course materials and class exercises
Define instance requirements
Define classroom computer set-up requirements
Train-the-trainers
Populate training seed data
Determine an instance cloning and maintenance schedule
Determine a classroom computer maintenance schedule
Continue to communicate training plans to all personnel

Appendix 1: Sample Work Plan for Implementing the Training Lifecycle for End-User Training

Test & Integrate Training Courses and Materials

Review courses and materials

Deliver pilot class

Integrate recommended changes into the class

Conduct final review and class pilot

Deploy Training

Schedule the training

Communicate the schedule

Deliver the training

Ensure Training Effectiveness

Gather data on a variety of scales using proven assessment techniques

Analyze data to determine the effectiveness of the training in regards to meeting user needs, system or process needs, and/or training objectives

If a deficiency is found, an issue is raised and addressed appropriately

Deliver new or revised classes as they are developed