How To Do Performance-Based Training

Training Application Notes

Purpose

This document explains the jTask approach to developing and teaching Performance-Based Training courses.

Overview

The objective of Performance-Based Training (PBT) is to teach students how to do the tasks required by their jobs and to confirm that they can do those tasks, in an effective and time-efficient manner, when the training finishes.

The content development process for a PBT course begins with Job Task Analysis — the determination of the Job Tasks to be mastered and their Key Results, which are the standards to which the tasks must be performed. This establishes the evaluation criteria for the Skills Tests. After the completion of the Job Task Analysis, the course components are planned out — the Skills Tests, the Practices, the Job Task Demonstrations, and the Theory to support the Job Tasks. Then, and only then, do we set about producing the actual material of the course components. You should note that this development order is important. By establishing our testing and teaching goals first, we establish the constraints and boundaries for the course material. This ensures that we develop the material we need and no more, which in turn keeps development cycle times to a minimum.

Of course, when teaching a PBT course, we approach the material in the reverse order. We start with Theory to build a foundation of knowledge and work our way up through the Demonstrations and Practices to build performance skill, as demonstrated through the Skills Tests and certification.

In the following sections of this Application Note, we will take a closer look at what is involved with determining Job Tasks and Key Results, developing all the components of a training course, and delivering the course material to students.

Guidelines

How Do You Develop Performance-Based Training?

Job Task Analysis is the heart of Performance-Based Training. It sets out what a person needs to be able to do by the end of training. Or in other words, what a person needs to be able to do on the job.

So the first step in developing PBT classes is determining the Job Tasks our target student audience is expected to master. As

Example: Command Verbs

- **Clean** the floor
- **Install** the battery in the radio
- **Align** the laser to the system's optical axis
- **Test** the personal computer's "motherboard"
- **Replace** the spark plugs in the engine

Tip 1: Use command verbs for better comprehension.

a part of this activity, we work not only to identify individual tasks, but also to organize them into logical Task Groups and develop Objectives for those Task Groups. This allows us to work out how the tasks are used together on the job to achieve larger aims. Additionally, we consider how the Job Tasks should be certified. If they are important tasks or involve critical safety issues, we want to make sure it is understood up front that students must individually perform the tasks to be certified, as opposed to being certified through oral, written, or team testing.

The second step is to determine how well the students should do the Job Tasks — that is, what the Job Tasks must accomplish, the Key Results.

For example, say we wanted to define the job of a web designer. An example Job Task and its Key Results might be:

• **Task**: Create a default home page

• **Key Result (Standard)**: The home page file is named "index.htm"

• **Key Result (Standard)**: The home page file is saved in the correct directory

• **Key Result (Standard)**: The web server start up page is set to "index.htm"

• **Key Result (Standard)**: Browser detection javascript is added into the header tags

• **Key Result (Standard)**: The META content and name parameters are defined within the header section

Or for example, say we wanted to define the job of a bicycle courier. An example Job Task and its Key Results might be:

• **Task**: Cycle 5 miles (16.09 km)

• **Key Result (Standard)**: 20 lbs (9.07 kg) are carried

• **Key Result (Standard)**: Distance is cycled in less than 16 minutes

• **Key Result (Standard)**: The course cycled is the route between Union Square and the Mission District

You should note the specificity of the Key Results and that they answer the question, "How do I know I'm *really* done?" In the first example, we could have simply said "The home page is created" — but this would not ensure that our file naming rules were followed or that critical code was always included. In the second example, we could have said simply "5 miles cycled" —

but this would not ensure that our couriers would train to cycle on the kind of terrain that exists between Union Square and the Mission District, or that they could carry 20 lbs and complete the run, or that they could reach their destination in less than 16 minutes.

Step three is to plan the course components needed to teach the students to perform the Job Tasks and achieve the Key Results. That is, step three is to determine how many Skills Tests, Practices, Demonstrations, and Theory elements are needed and what they each should cover.

Our work from step one to organize the Job Tasks into Task Groups often proves very valuable at this point. Task Groups frequently represent a natural structure for a course, and their Objectives suggest the context within which the Job Tasks must be practiced and what background knowledge (i.e., Theory) is needed.

When planning the course components, we do so in a specific order, considering first the Skills Tests and then the Practices, Demonstrations, and Theory. This is because it is, as with the overall development process, important to understand our ultimate goal. Knowing what we want our students to be able to handle in terms of the Skills Tests, we can then figure out what kind of Practices, Demonstrations, and supporting Theory are needed.

Next comes step four, the actual creation of the course components. Many experts in instructional systems design suggest writing the course components in the same order that they are planned. However, this is not necessary. The writing process for each course developer is always unique, and many developers find it more workable to develop the material in the order it will be taught. Regardless, though, it is almost certain that the process will be iterative and that "completed" components will be tweaked here and there throughout the development process.

Useful things to keep in mind when creating the course components include:

• **Skills Tests**

It is true that our primary concern is the ability of students. And it is also true that we define the method of certification (oral, written, team performance, or individual testing) during the Job Task Analysis stage of development. However, it's important to remember that we always have the leeway to do additional testing. At times, we may want to verify that students have mentally grasped certain aspects of a task, which are best confirmed through oral or written testing. Examples of such things include interpreting system software messages or choosing values for system parameters.

- **Practices**

  To be effective, your Practice sessions must allow students to experience, not just the activity of doing the Job Tasks, but the impact of doing those tasks under different conditions as well. Practice is when your students should get the chance to explore the limits of the Job Tasks — like the difference between adjusting a part one way and then another.

- **Demonstrations**

  This is where the students get their first glimpse of how a Job Task is done and start formulating how they, themselves, would do it. This is also a place where you have an opportunity to reduce training cycle times by pre-preparing demonstration stages of performing the Job Task. A classic example of pre-preparation is on cooking shows, where a recipe is prepared several times in advance, to different points in the process, so that the cook doesn't have to do all the chopping, marinating, etc. on the spot. In an instant, they can throw the ingredients in a pan, put a pan in the oven, and pull out a finished product to demonstrate the desired results at each stage.

- **Theory**

  Not to be forgotten as a part of the Theory is information on why and when a Job Task must be done, as well as infor-mation on what it involves. But note, it's important to limit your Theory content to what students really need to know. If you let it run away with you, Theory will take over the entire the course. Developing the right amount of Theory is often one of the biggest challenges when putting together a course, especially in highly technical environments. Frequently, the temptation is to "leave no stone unturned," but this leads to unending development cycles and a dearth of hands-on practice due to excessive amounts of Theory. If your material is extensive, however — or even if it is not — you may want to consider packaging this part of your training as "home study," to be covered prior to the arrival of the students on site for their training. In this way, students can come prepared and you can maximize the hands-on practice time during class.

Last but not least, the final step in the development process is to test your course materials once they have been completed. Testing your course materials is very different from reading and reviewing the material, and should be done by actually running a pilot class. A pilot class may be thought of as a kind of dress rehearsal. It allows an instructor to work through details and "iron out the kinks" in moving students through the courseware. As you would never open a Broadway musical without rehearsing your cast, so should you not put students into a never-tested class. The next section, "How Do You Teach Performance-Based Training?" discusses the teaching process that pilot classes should examine.
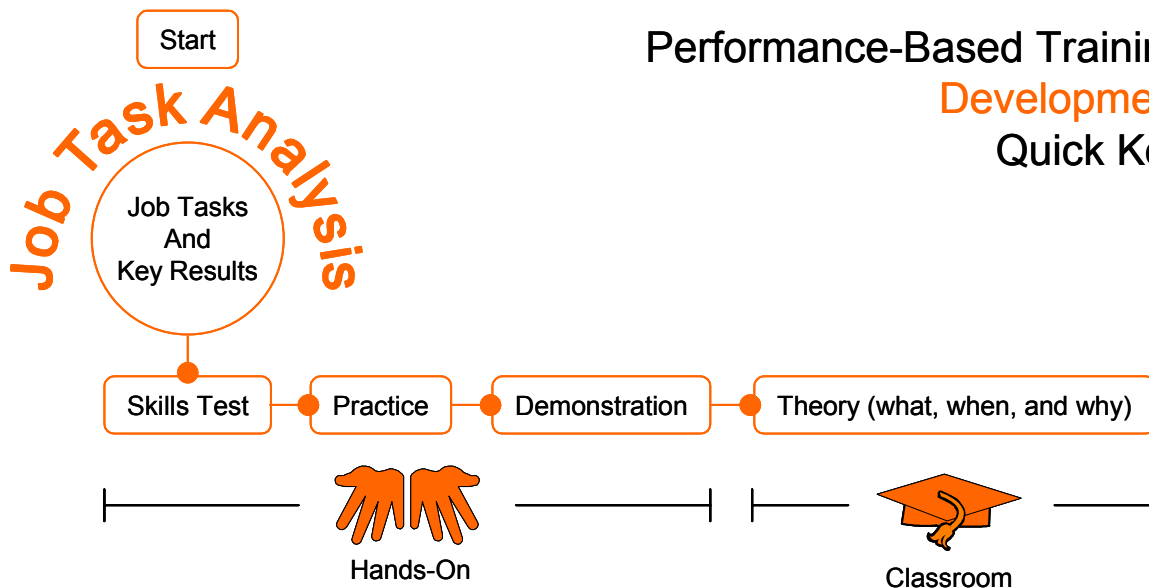
Figure 1: When developing PBT courses, you develop the Job Tasks and Key Results to be taught first (your goals), and then the Skills Tests, Practices, Demos, and, finally, the Theory.

How Do You Teach Performance-Based Training?

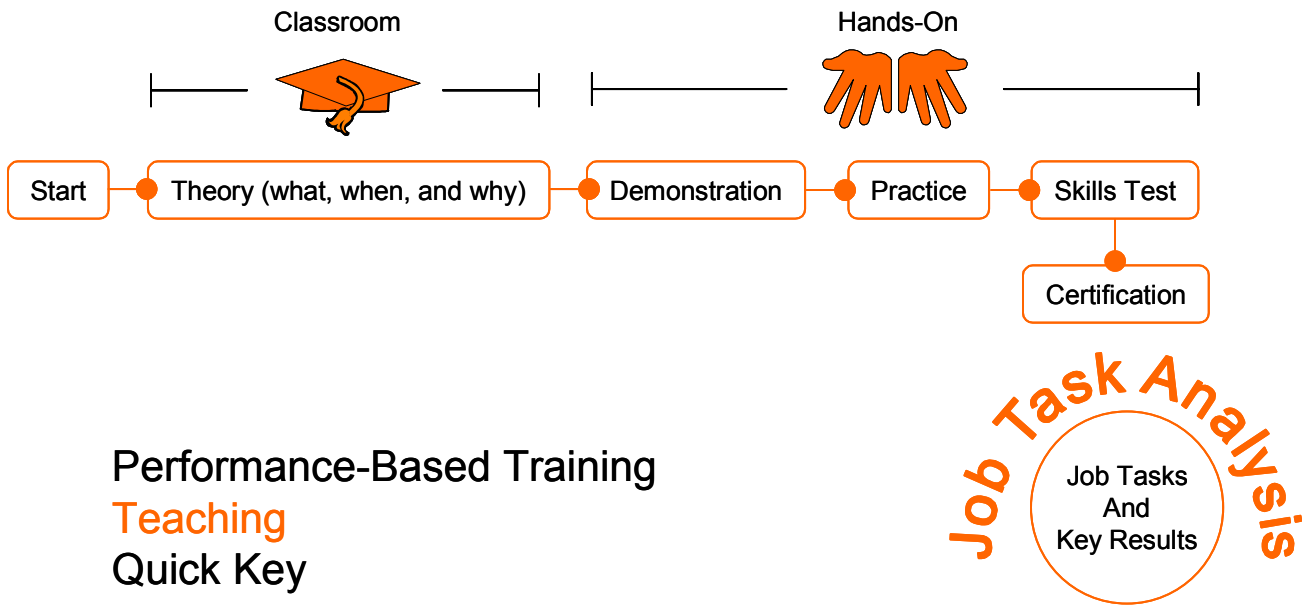Teaching Performance-Based Training is done in the reverse order of the development process.

As we noted before, we lay an initial foundation of knowledge by starting with the Theory behind the Job Tasks. The Theory can be delivered as lectures or as pre-reading material, and the goal of this part of a PBT course is to cover the "what, when, and why" information for the Job Tasks. For example, if we were conducting a course on replacing laptop computer LED screens, our Theory information might include such things as an explanation of how a screen is connected to the main portion of the computer, problems that require a screen to be replaced, and an overview of the replacement process.

Following the presentation of the Theory, the instructor might demonstrate how the Job Task is done. Demonstrations are not always required, but can be useful, as they allow students to prepare themselves by mapping out complex task steps mentally. In our example, a Demonstration of replacing the computer screen might include best practice methods for detaching the screen and how to take important safety precautions.

Once "armed" with the Job Task Theory and the knowledge of what to expect, as shown by the Demonstrations, students then proceed to the Practice phase of the course. The students practice until they have mastered the Job Tasks. Then, all that is left to do is to confirm that the students can do the tasks. Written tests can be useful, but observing individual students perform the tasks is the preferred method of confirmation (certification).

It is important to note that you do not have to make a student go through all steps when doing PBT. If he or she knows that they just need to practice and then take the test, then that is permissible. Also, if it is recognized that a student can already perform the required Job Tasks, they also should not be made to go through the outlined training courses. PBT says, *training is finished when a person has mastered the Job Tasks, not when a course ends.*

Figure 2: When teaching a PBT course, you present the Theory first and follow it with Demonstrations and Practice opportunities. When the students have mastered the Job Tasks in their Practice sessions, they take the Skills Tests to prove their ability and become certified.

A Caution

Much training goes wrong before it is even started by defining the length of a class before doing anything else. And as a matter of fact, reducing the length of whole training programs is a very common way companies try to reduce training cycle times. But starting course development with limiting class length as your primary goal presents two fundamental problems:

1. The training content will be defined by what can be taught in the time available, not by what your students need to be able to do.

2. Practice time will also be defined by the available time, as opposed to the time needed to really learn the required skills.

This means that, at the end of a training course, it will be highly unlikely that the students will be able to do the target job when they complete the course.

Of course, time is a factor that must be weighed against other business constraints. If all else fails, make sure your training covers the critical tasks and focuses on hands-on practice by sending the Theory in advance.

A Final Note

This methodology for developing and delivering Performance-Based Training has a number of advantages.

1. It provides a clear organizational structure for your courses, which allows you to prepare training time requirements with greater ease, as well as show the impact of adding more students on equipment requirements and course length.

2. It protects your company's investment in training. With clearly documented course components, new instructors can assume responsibility for courses more quickly and with greater ease.

3. It is simple to understand and crystallizes the instructional design process for non-practitioners.

4. It improves student satisfaction, since it enables you to provide a clear description of the service provided by your training department. Through the Job Task Analyses you can create meaningful course descriptions and set reasonable expectations.

5. It allows you to use pre and post self-assessment to measure actual performance results — as opposed to the traditional "smiley sheet surveys," which only reflect whether or not students enjoyed the time they spent in class.

Table 1: Training Cycle Time for *N* Students

No. of Students	Theory	Demonstration	Practice	Skills Test (Certification)	Total Time
1 Student	2 hrs	0.5 hrs	4 hrs	2 hrs	8.5 hours
3 Students	2 hrs	0.5 hrs	12 hrs	6 hrs	20.5 hours

Figure 3: The above tables show how time requirements can be better estimated with a clear course structure. Row 1 shows the Training Cycle time (Total – 8.5 hours) for one student with one set of equipment. Row 2 shows how the training is impacted with three students but still only one set of equipment (Total Training Cycle time 20.5 hours).
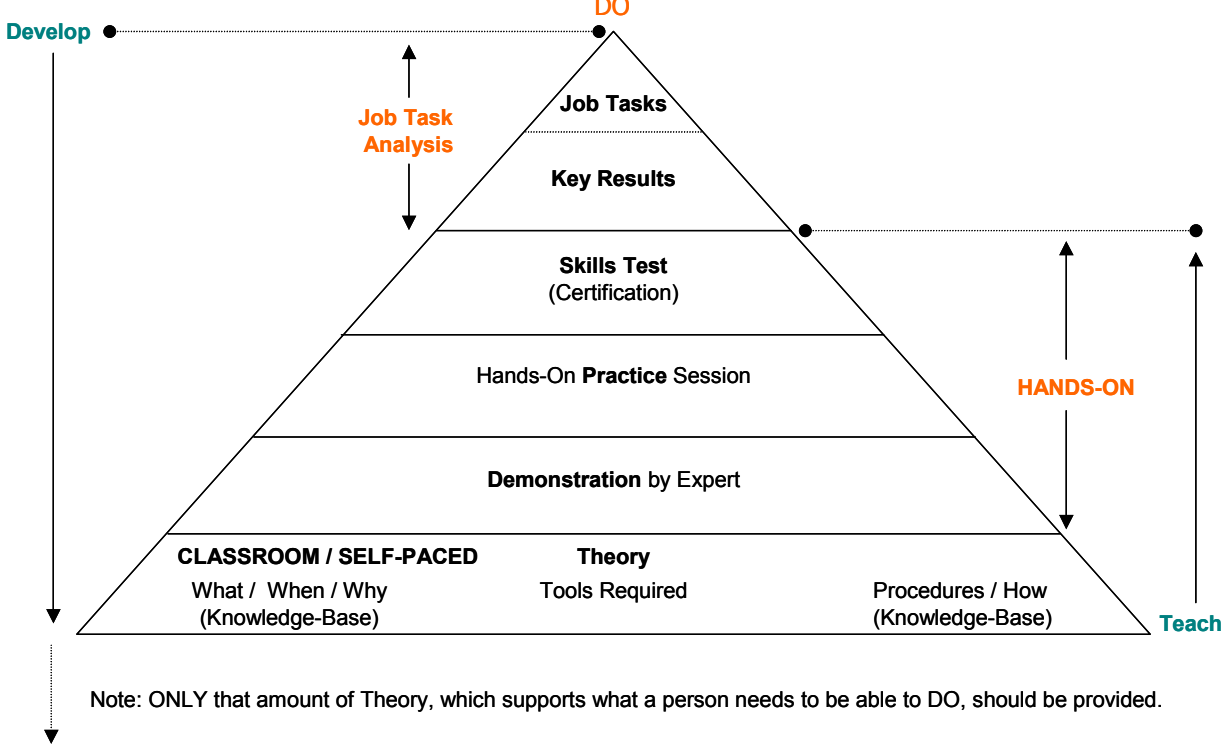
PERFORMANCE-BASED TRAINING
Quick Key

Figure 4: This figure gives an overview of how Performance-Based Training is built and taught.