



Live-File or Play-File? What Is Best for Training Project Management Software?

By [Volker Bendel](#)



When implementing job costing and accounting systems and planning the end user training, a question frequently raised is: "should we train our users on live data or should we create a dedicated training file?"

There are arguments for either solution:

Having a "training" (or "play") - file will give users the security to be able to try out entries, knowing that this will not affect the immaculateness of their live data. It will allow to enter any conceivable scenario and correct any possible errors that one would not want to enter into a live system just for training purposes.

Based on samples, provided for the training, users can learn any imaginable job costing situation on a blank canvas, giving them the opportunity to see 100% of the functionality their new system has on offer.

They will come out of the training with the peace of mind, that whatever they have tried out just now will not have "contaminated" their Live data when they log on to that for the first time.

The other school of thought would argue against using a play file:

It is important for users to see their new software as a tool for their day-to-day work and not a toy to play with. During a system changeover time for clearing off data in the old system and getting everything ready for the new system is at a premium. Why spend this valuable time in training seeing functions that will never be used for real work as they are not relevant to how that particular business is run? If trainees instead practise on their live data entering live information, much better use can be made of the training time. The agenda is not based-around but is with real life cases and attendees will come out of the training session and have actually done work rather than looking at made up scenarios, that a person outside of their business (the trainer) has prepared for them to enact. The same applies to mistakes: mistakes are eventually going to be made when the system is live and if made on live data, it is better for those to take place during a training session as the trainer will be able to show how to rectify them and the trainees get the reassurance that every error can be corrected. Mistakes made during a training will be the ones most common to happen after the training and it is more important for users to remember how to put them right than to see what other potential mishaps there could be and how to adjust those. If users come out of their live training confident how to correctly use their software and that there are solutions for anything that may go wrong the training session is of much more benefit to them.

Experience from many implementations shows that the second approach is the preferable one:

There is an improved acceptance of the necessity to attend the training because users understand that the training work will be real work on live projects. It will also be easier for trainees to memorise what they have been trained on as it will still be there in their database. Following on from the training that covered real jobs users are able to base new jobs on the templates of those worked on during the training rather than having to access a "play file" to look at an example in that file from the training and re-enter that into Live.

Experience also shows that "Home Work" and "Practise Work" users are asked to undertake in between training sessions is on average only completed by 5 to 10 % of users accessing a training file. They just can't justify to spend time practising on training jobs, whilst real work needs to be completed. In contrary around 60 to 65 % of users who had training in a live environment will have worked with their new system prior to the follow-up training class as they will try to apply what they learned working on one live job during the training to other live jobs.

All in all companies have found that they get a much higher return on investment from their training bookings in implementations involving "Live training" rather than "Play training".

© 2013 Volker Bendel - Volker Bendel is head of the Project Management and Training Department of Agency Software Worldwide, the producers of the "Paprika/Rebus" job costing software (<http://www.paprika-software.com>)(<http://www.rebus-software.com>). Originally from a legal background, he has worked with the Creative Industry for over a decade implementing Job Costing and Accounting Software Systems. He has also delivered training courses in the UK, Europe, the UAE, the US, China, Singapore/Malaysia, Japan and Australia. Prior to that he worked as a senior business consultant in Hong Kong and as a department manager of a design department in Hong Kong.

Article Source: http://EzineArticles.com/?expert=Volker_Bendel