Assima
Innovate & Accelerate

Alternatives to Failing eLearning

Why software simulations are not good enough for end user training and performance support

Author	Eric DUNEAU
Date	12/05/2009
Version	1.0
Audience	All

Overview

This whitepaper reviews the reasons why eLearning and software simulations have progressively replaced more standard training methods, and why they failed to live up to expectations. It also reviews alternatives which have the advantage of delivering such expected results and overall returns on investment.

Table of Contents

Executive Summary

Abstract

This document starts by examining studies and facts that have been published on Systems Training and looks at why training therefore is both a necessity and a desperate cause. It examines what makes up the ideal training environment and the risks for bad or no training.

It then looks at why substituting training for eLearning is bound to fail and why software simulations are not fit for purpose and fail to replace adequately the ideal training environment. It covers the issues of maintainability and reuse and examines whether simulations have a place at all.

It next examines an alternative to eLearning, in-application performance support. It gives a brief review of performance support systems tries to understand end user expectations.

It then describes Assima's proposition including replacing the training environment with software clones, instant in-application performance support and measuring ROI.

Key Terms

Throughout this document, several key terms are used regularly.

The first time a term is used, please refer to the Glossary section at the end of this document to ensure you fully understand the concept presented.

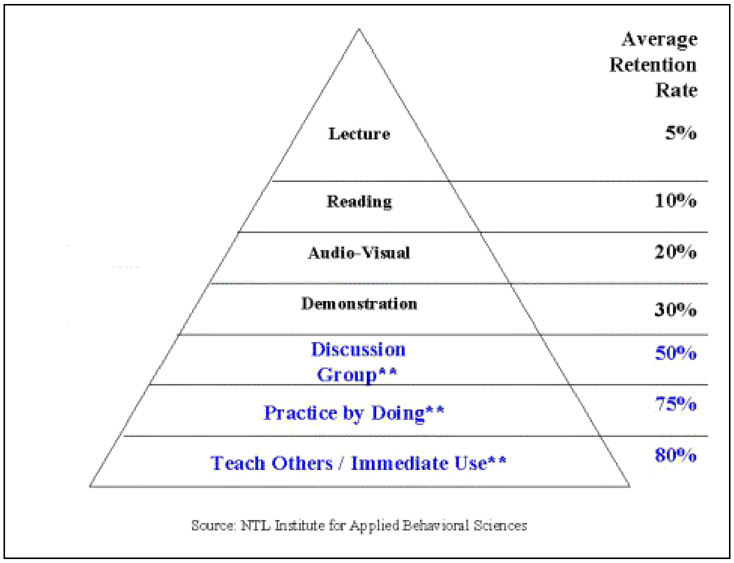
Systems Training: Reality and Related Expectations

Studies and Facts that make Training both a Necessity and a Desperate Cause

There are several key facts that are fundamental to successful Systems Training. This section exposes facts and studies which have been validated, sometimes dozens of years ago, by eminent research. These studies are unbiased, not linked with any simulation software, and for the most part, have been done before simulation software even appeared on the market.

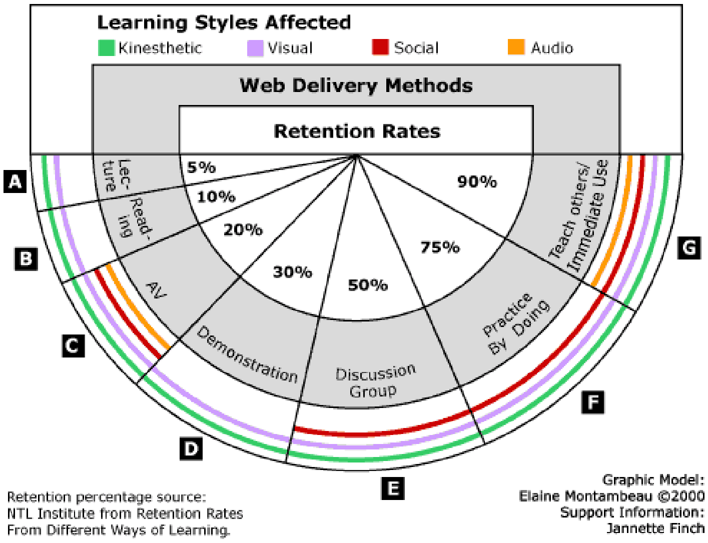
End user retention rate vs learning activity

study



Source: NTL Institute for Applied Behavioral Sciences

Figure 1: Taken from http://www.usdla.org/html/journal/APR02_Issue/article01.html



Retention percentage source:
NTL Institute from Retention Rates
From Different Ways of Learning.

Graphic Model:
Elaine Montambeau ©2000
Support Information:
Jannette Finch

Figure 2: Taken from http://www.cofc.edu/bellsandwhistles/research/retentionmodel.html

The end user retention rate varies considerably depending on the implication of the end user. Achieving a 75% retention rate is only possible through practise by doing / by error, which forces the user to learn, understand, and remember.

As the study proved, this fact is part of human nature, and cannot be changed by modifying behaviour. It has to be taken into account when delivering end user training.

End user retention vs time

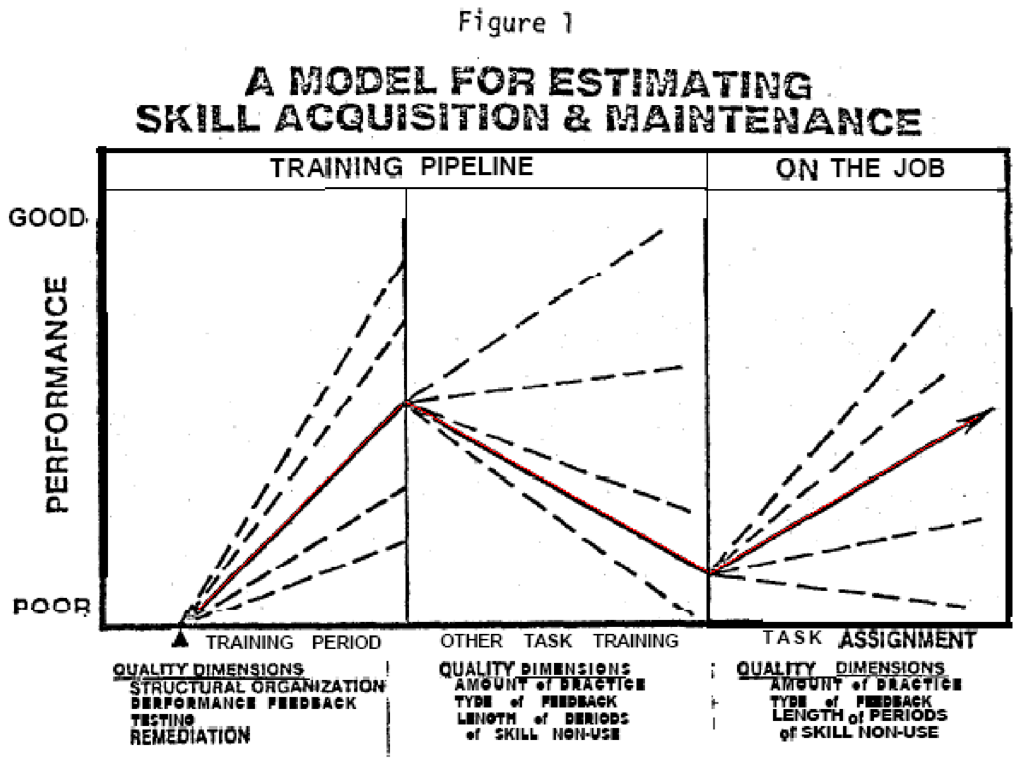


Figure 3: Taken from http://www.internationalmta.org/Documents/1983/Proceedings1983.pdf (more on http://abe.villanova.edu/proc2003/kamuche.pdf)

End user retention falls dramatically after days and weeks of non practice. It then goes back up when doing the actual tasks in production, with the hope of matching what was initially learnt and forgotten.

End users learn more by making mistakes and understanding the reasons for them

We will just illustrate this with a simple quote by Robert Kiyosaki, well known author and accelerated learning expert, which sums it up very nicely:

"Our education system teaches riding a bicycle by lecturing on the subject for fifty hours, giving a written test, and then punishing the student for falling off"

Basically, if you are not put in a position to learn by experience, by actually doing it, and learn by making mistakes, understanding them, and fixing them, then you will just not be prepared and will not have the required skills. Mistakes tell you that you do not know something that you need to find out; hence you improve dramatically your retention ratio.

Training by only showing the best practice, and without leaving the possibilities of making mistakes, is much more like a demonstration (30% retention maximum), whereas learning by doing more than doubles this retention level (75%).

For more scientific research and evidence, see "Learning from Mistakes" on this website http://www3.interscience.wiley.com/journal/118582763/abstract?CRETRY=1&SRETRY=0

Training is important for end user confidence and end user adoption

study

Seven out of ten UK government projects fail despite a £14 billion annual spend on IT.

[May 07, http://news.zdnet.co.uk/itmanagement/0,1000000308,39287110,00.htm].

This figure is estimated to be similar in the private sector with the Standish Group maintaining that only 34% of all IT projects are successful. AMR Research found that even amongst successful CRM implementations, 47 percent of companies reported serious challenges with end-user adoption that often put projects in jeopardy.

[http://www.crm2day.com/news/crm/EpupEEFuAlbfuEWzQS.php]

Training is not just important for the sake of improving end user skills. There is a fundamental element involved too: achieving end user adoption of the system itself. In the end, the system in place (ERP, CRM or other enterprise application) will only be as good as how the end users use it. If users reject it, it becomes worthless. If users make the most of it, it can bring returns on investment even larger than anticipated, and accelerate the company's performance.

The Ideal Training Environment and Associated Constraints

The ideal training environment

Taking into consideration the facts and studies examined above, it becomes very clear what the ideal training environment should consist of:

- Access to the real application on a training database
- Access to a personal coach / trainer
- Opportunity for lots of practise and coaching

These are the conditions which will guarantee, as much as possible, the best transfer of knowledge, leading to high end user retention, and also high end user adoption. This is what has been delivered consistently pre-internet area, with relatively good results.

Problems with the ideal training environment

There are several constraints associated with this ideal training environment:

- High cost:
    - The cost of licences used just for the training sessions
    - The cost of setting up the database, and the cost of maintaining the training data to ensure it is up-to-date
    - The cost of maintaining the training infrastructure (software, hardware, logistic, rooms etc…)
    - The cost of moving people off work (non-productive time)
    - The cost of moving people to training centres (travel, accommodation, expenses, etc…)
- High risk
    - The risk of the training storyboard not working (wrong data, application has changed, etc…)
    - The risk of the network not working (no access to the training environment)
    - The risk of the trainer not being available (risk on quality of the training delivery)
    - The risk of user retention being low and eroding quickly with time

Furthermore, it is not practical for training new intakes, or small populations, or refreshers…

Understanding the Risks of No or Bad Training

Because training can be seen as costly and its value or impact on the company performance is difficult to measure, it is very tempting to minimize it or even bypass it completely. The consequences may not be felt immediately, or may not be that measurable if no feedback loop is put in place. However, we can sense that it certainly creates additional risks in terms of end user efficiency, and possibly end user adoption.

Depending on the complexity of the application and its processes, depending on the number of end users and their initial skills, it may be that the lack of training passes unnoticed, or on the contrary, it could negate all efforts spent on deploying the systems in the first place, which is generally measured in one or two orders of magnitude above the training costs.

So, although it may be possible sometimes to minimize the training efforts for the right reasons, the right analysis, it can also potentially have disastrous effects on the company's performance if it results in weak end user adoption, or even a rejection of the system. It is important to note right now that a "bad training environment" could have the exact same disastrous effect as no training at all, or even worse, in terms of user adoption / rejection, and cost of errors (process time to redo, client value, financial loss, etc...).

Substituting Training by eLearning: Obvious Reasons for Failure

Software Simulations are Unfortunately Not Fit for Purpose

The need for an alternative to the ideal training environment

The explosion of systems used in large companies, the increasing number of end users, and the need for reducing costs and risks have led to the search for alternatives to the classical training approach. The Internet, and the ability to dematerialize and virtualize the training environment, have pushed towards using a new medium for delivering systems training, and the birth of "eLearning".

However, it has not been properly recognised that this replacement of the training environment by new eLearning materials is coming very short of the ideal training environment, and its associated qualities.

It seems that democratizing training and cutting costs were the main drivers, leading to the said training solutions coming very short of expectations, and the subsequent birth of a second wave branded as "blended learning". It is also likely that so long as blended learning solutions do not learn the lessons of the failure of eLearning, they head towards the same failure, the additional cost on top.

What are software simulations anyway?

A flight simulator generally reproduces entirely, with high fidelity, the real flight commands and experience, using an exact reproduction of the pilot cabin, all buttons, knobs, gauges, software, and plane behaviour acting in real time.

Unfortunately, a software simulation is generally a pale reproduction of the real application, being a mere succession of screen copies which can be clicked upon in specific designated areas, or hot-spots. For this reason, the name "software simulation" is very misleading and led people to jump to conclusions and expectations way out of its reach and capabilities.

For example, the standard user retention associated with usual software simulations, in all their commonly accepted publication modes (from show me to test me), would be more in the region of 10-20%, as they lack a lot of the interactivity and freedom of use that make the user think and learn.

Why simulations fail to replace the ideal training environment

It is a fact that software simulations can lead to excellent score during training, such as 90% or over. However, the big flaw is that nearly anyone could get such 90% or over, on a simulation written in Chinese or Korean, despite not understanding a word of it, as the clickable hotspots, red circles and mouse over icons will undoubtedly guide to the right action. The conclusion of such an experiment would show that it is possible to get all users certified after taking eLearning courses, but their understanding of the concepts and their retention would be near 0%. This eLearning material is clearly not fit for the purpose of end user training, especially when the goals of end user adoption and end user proficiency are the main drivers.

Furthermore, when comparing the software simulations capabilities with the studies concerning software training, end user retention, and end user adoption, it is very clear that the said software simulations come very short of the minimum requirements. If you cannot get passed 20% retention rate for a start, forget getting any subsequent benefit from these simulations. They are simply not fit for purpose, and shall in no case be used for end user training. In this situation they risk being more damaging by lowering end user adoption as these users receive very non engaging eLearning content, which is generally their first experience of what the real system will be a few weeks later.

More Hidden Trouble Ahead…

Untold story of the (un) maintainability of software simulations

In the last 10 years, the hype has been about who can develop software simulations faster. Contests have been organized in a Hollywood-type show every two years (Brandon Hall being one of the most active in these types of events), to elect the best productivity tools, based on 15mn shootouts to create as much eLearning content as possible, including software simulations. These events can be fun, and focus the mindsets on the grail of faster and faster training content development, but they totally miss the point. Indeed, every single tool on the planet generates the said software simulation within less than 10 minutes in these events. But the real grail lies elsewhere: quality of content and maintainability of content. What is the point of generating software simulations in the blink of an eye, if they are not good enough to be used for end user training, and have a life span of a few days at best? Most are simply un-maintainable. To cope with the constant change of the real application, they need total wipe out, recapture, redesign, in short: a complete redo.

Because of the pressure to deliver new applications as they come out of customization by IT, these eLearning software simulations developed in the fastest pace are still very often outdated even before being used for the first time. The application has changed… and the simulations have not been recaptured as it was not possible within the timeframe.

There are two main reasons for this:

- the application will generally change until the very last minute (ie roll out time)

There is nothing you can do about it. It is a fact of life that to allow enough time for capturing 100 best practises in software simulations, requires one or two months of work, and during this time, the application’s UI will change 10 times. The screen copies making the simulations are already outdated a few days after being captured.

- 80% of the hard work is in the best practice design, however 80% of the reason for change are in the last minute changes of the application UI

In most cases (we are talking over 90%), instruction design is placed directly on top of the screen copies. This instruction design represents the hard work, the several hours spent on making the simulation look good, so that some kind of knowledge transfer to end users occurs. This is as opposed to the 10 minutes spent to capture the 20 screen copies per simulation (as the shootouts can prove). However, because instruction design and screen copies are intrinsically linked, binning the later for recapture means all the hard work goes into the bin as well. The training content therefore becomes highly un-maintainable with more volume and pressure on deadlines. And the only compromise is either to lower the quality of the instruction design, making these simulations even less fit for purpose, or release them despite being de-synchronized with the real application.

In conclusion, it is very important to understand that software simulations are highly un-maintainable, and are likely to be outdated before even being used.

The content exists, let s reuse it everywhere

The idea that some content made for training purpose (even if it is a long way from achieving this goal …) could be repurposed for some other needs, such as Performance Support, is flawed from the start. Before reusing the content because "it's free", one has to look at the users expectations when faced with skills or knowledge gaps in production. In most of the cases, the user can be characterized as being mostly impatient ("I need an answer to my problem right now") and mostly intolerant ("Give me the right solution to my problem right now"). In most cases, software simulations will not satisfy impatient and intolerant users. As was the case with replacing the training environment by software simulations, reusing them for Performance support is another mismatch.

What shall we do with these Simulations then?

If software simulations are not fit for purpose for either training or performance support, what is their benefit?

Apart from the demonstration mode, which has some role to play in end user adoption and performance support when well designed and small enough, there is not much to gain from the other output modes of the such simulations. They are indeed not fit for purpose, and although they may seem to come for free as additional output from the initial capture, the good advice is to not use them.

An Alternative to eLearning: In-Application Performance Support

What is Performance Support and Why is it Necessary?

A brief review of Performance Systems (or ePSS) attributes

A commonly accepted definition of an ePSS (electronic Performance Support System) is:

"an integrated electronic environment that is available to and easily accessible by each employee and is structured to provide immediate, individualized on-line access to the full range of information, software, guidance, advice and assistance, data, images, tools, and assessment and monitoring systems to permit job performance with minimal support and intervention by others"

As the broad definition indicates, ePSS systems have been around for quite a while, under one form or another. The important notion behind such system is really "to permit job performance with minimal support and intervention by others". However, as software is becoming more and more complex, it is very important to keep in mind that a type of ePSS which was proving successful 10-15 years ago, may just be worthless nowadays. As the notion of "job performance" gets more complex and demanding by the day, the systems that allow it must also adapt and be smarter.

End users forget whilst applications become ever more complex

As seen in the study about end user retention vs time, users tend to forget acquired skills if they do not practise regularly. But another fact is happening in parallel, which is the consequence of the Moore's Law (processor power doubling every 18 months). The consequence is that as computer power increases, it leads to more and more complex operating systems, more and more complex user interfaces, and more and more complex applications - to the point where more enterprise processes become digitalized, and that explosion of complexity creates even more challenges for end users. We can summarize this with the following illustration:

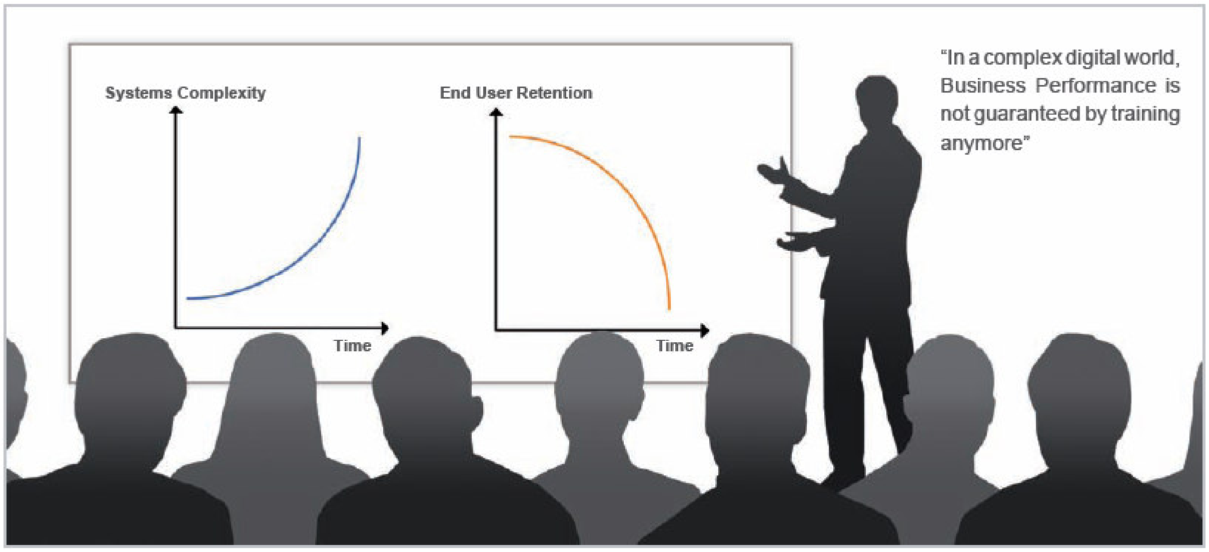

Systems Complexity
Time
End User Retention
Time
"In a complex digital world, Business Performance is not guaranteed by training anymore"

Figure 4: Systems Complexity vs End User Retention

This new challenge creates an even bigger one: it is likely that end user training is not going to catch up with this challenge, as more training will be required, on even more applications, more complex tasks, and with a high risk of retention falling fast after the training event.

End users are smarter and more educated

Despite the gloomy and desperate situation described above, there is hope. As complexity surrounds us, people have been forced to improve their skills, just to deal with the complex world around them. People are more educated, smarter, and the trend is here to stay with the internet generation (Generation Y). But at the same time, the way this new workforce learns and gains skills is different. Training takes a part, but self training, research, and the ominous internet searches is more and more prominent as a method of learning. We have more and more smarter people, who learn just at the point of need. They do not need to remember or acquire skills. They just need to know how to access the information at the right moment, and they excel at that.

The smarter generation Y is therefore better prepared for an ever changing and complex world, and an ever changing and complex information system. But the other side of the coin is that this new workforce, when put under pressure to deliver through digital systems, will be highly intolerant and highly impatient. They accept they do not know all the answers because it is just not possible, but they will not accept that they are not given ways to find the answers by themselves quickly and efficiently. They require in-application performance systems which help them be proficient without being experts.

You do not need experts, you just need efficient users

Training is a good way to transfer skills and ensure end user adoption. But if there could be other ways to transfer skills without impacting the performance, and to ensure an equivalent end user adoption, then this would be as good as training, maybe even better because it requires less off work activity.

This is what in-application support is about: ensuring users are proficient, and delivering the right amount of information, at the right time, when they need it to stay at the same level of proficiency.

It is very important to note that in-application performance support is not the ultimate solution that replaces all training requirements. It is a complement that plays a big part in the end user adoption / end user efficiency battle on ever growing complex information systems.

Understanding End User Expectations

Reviewing the expectations for training

If we have to retain just one goal for using training, it would be certification. To certify that an end-user has acquired a set of defined skills and therefore can access the system and be proficient.

Certification generally takes hours, sometimes days to obtain. It is a long process of learning skills, preferably by doing and practising. The resources which are most appropriate for end user training are the ones described as the "ideal training environment" (training environment, trainer, and practise by doing).

The different expectations for Performance Support

The expectations for Performance Support are totally different. It is about giving end users the right information at the right time in an acceptable time period. Considering the end user is mostly impatient and intolerant, we may rephrase the expectation as "I need the solution to my problem in less than 10 seconds or I'm not going to reuse this Performance Support again"… "and by the way this Support System should know what my problem is when I press F1, as I do not want to spend time describing it".

With this in mind, do you think giving a training material to an end user asking for help is the right answer?

- I have a problem filling this field. Let's see what help I can get. F1!

- You can attend this lesson which will teach you all you need to know about XYZ. Pre-requisite: please spend 15mn so that we check you have the pre requisite for this lesson ; lesson duration: 1 hour.

It is obviously a NO GO! But strangely, this case is what is delivered to the user most of the time. You have a problem? Launch this software simulation which will show you how to do it. Most of the time, the software simulation will not give the answer to the problem as it will be mainly about how to navigate, not about business rules (the actual problem), most of the time it will take about 10mn to launch and review, most of the time it will irritate the end user.

Therefore, it is very important to understand that training and performance support are two very different ways of skilling up end users, and that they require very different material, infrastructure, and approach.

The graph below summarizes this concept that a good performance support (or ePSS) type of material has to be contextual and bring such relevance and readiness that intolerant and impatient end users will embrace it.

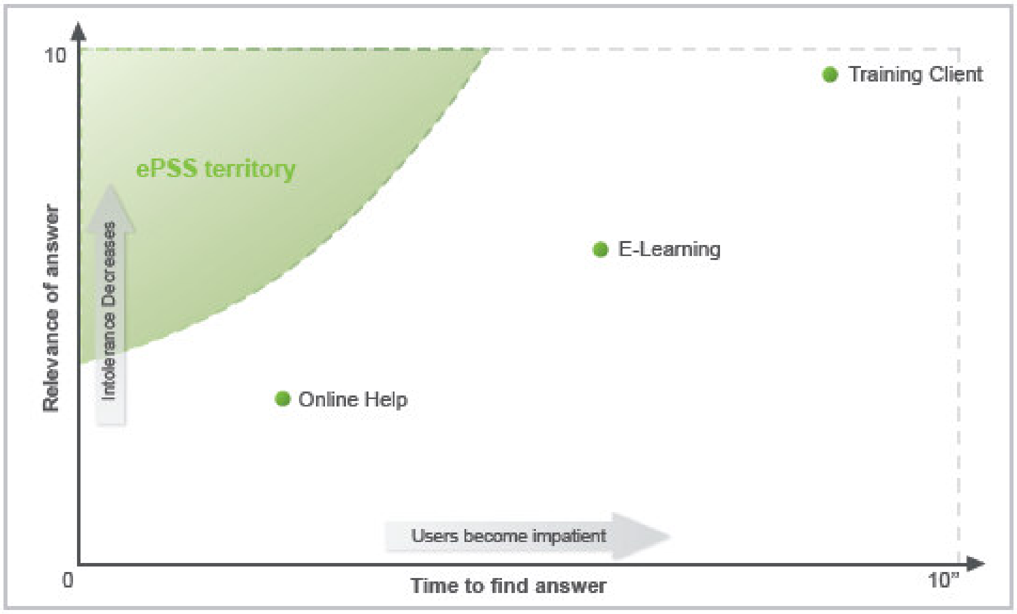


Figure 5: Relevance and Response Times when searching for answers

Different needs require different approaches

The following schema represents the very different expectations, needs, and solutions to put in place, for Training on one side, and Performance Support on the other.

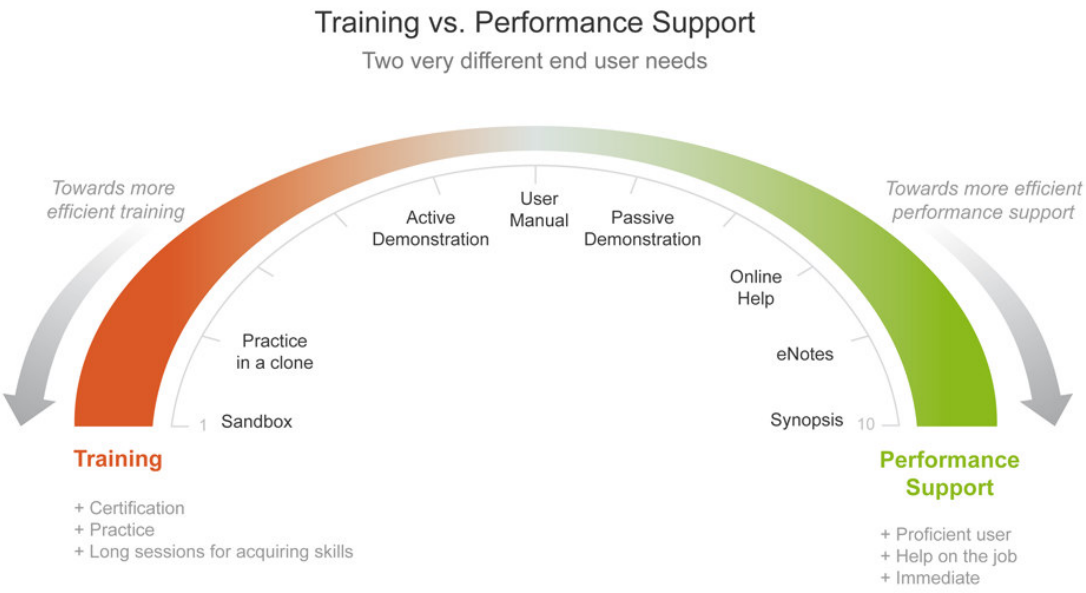


Figure 6: Resources used for Training vs Performance Support

The most important information to understand from this graphic representation, is that resources which help providing the most efficient training are typically the ones which are the least suitable for performance support. And resources which provide the most efficient performance support are the ones which are the least suitable for training.

Additionally, and as expected, software simulations are good for neither.

A Summarized View on Training, eLearning, and Performance Support

This is a quick summary of the view taken so far in this whitepaper.

- On Training

  Training is important for skilling up end users, end user adoption, and certification. Best end user retention is obtained via learning by practise, by making mistakes. Training takes time and cannot be expected to generate results in 5-10 minutes.

- On Performance Support

  Performance Support is more and more required to help end users perform their job efficiently, despite minimal advance training for the specific tasks to encounter. Performance support needs to be immediate and accurate (the right answer, right now, in context).

- On Software simulations

  Software simulations are not good enough to replace training environments, and cannot be expected to help deliver good training, with poor retention ratio of around 20% on average.

  Software simulations are also generally poorly adapted to Performance Support as they lack the immediacy and precision of the user needs, and focus too much on the best practice, not the business rules and all exceptions that will ultimately pose the majority of problems to the end users.

  Due to their usual low cost of production, software simulations have however been used massively up to now, for training and performance support of all types of software, including the more complex ones. This continuous use translates into poor user adoption, poor training experience, poor in-application support.

- On alternatives

  The low cost of production seems to have been the major factor in adopting software simulations en masse, despite their obvious low match with the actual needs. The minimum requirement for any alternative therefore has to have an even lower cost of production and maintenance.

Assima's Value Proposition

End User Training: Reviewing the Ideal Solution without the Costs and Risks

Replacing the training environment: software cloning

What if it were possible to keep the quality of the "ideal training environment", but at the same time get rid of most of the costs and risks associated to it? What if we could build a training client without requiring the licenses of such software, without requiring the database or the need to reset it, without requiring the network, the infrastructure coming with the full training environment?

By replacing the real application by a clone (a true simulation, in the sense of the flight simulator), and placing instruction design on top of this clone, it is possible to keep the same qualities of the real application and the trainer, hence to keep the same benefits drawn from the practice by doing, such as a retention rate near 75%.

If we now compare clone retention rate vs software simulation retention rate, and take into account the drop in retention after 2 weeks, an average lapse of time between training and use in production, we obtain:

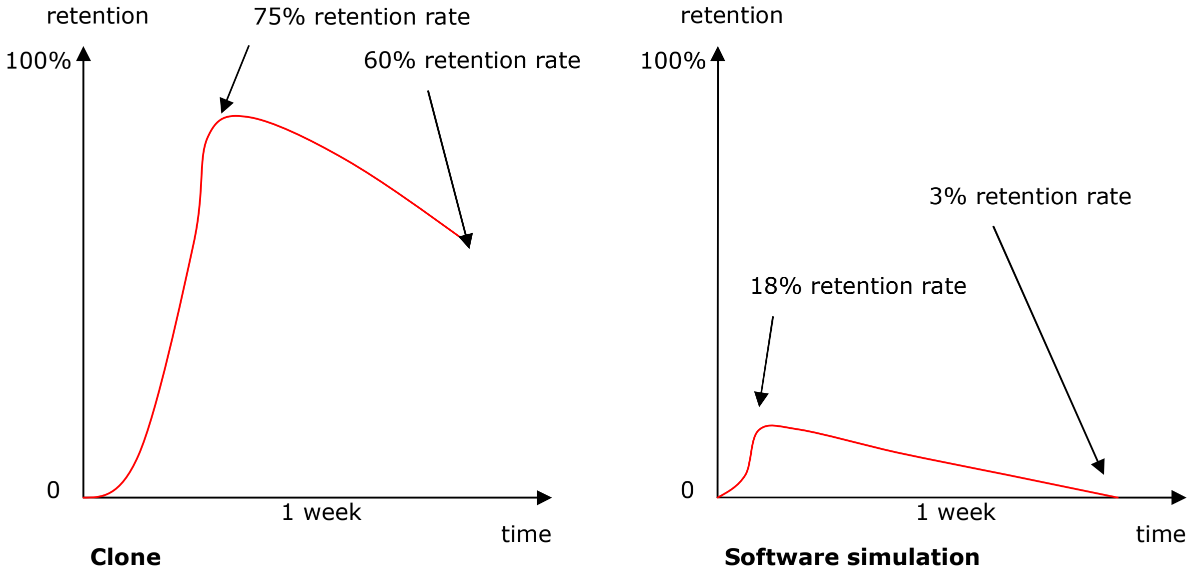


Figure 6: Retention Rates, Clones vs Simulations

This represents the difference between a fairly successful training session (60% retention at time of use in production), and a complete waste of time and money (less than 5% retention at time of use in production).

Why software cloning is the right alternative for the end user

Software cloning is the right alternative for these three main reasons:

- It can replace the training environment without diminishing the end user experience
- It can embed best practise, tips, and instruction design matching the ones of the best trainers
- It can be available any time, and any numbers of times, without requiring a specific setup

Why software cloning is the right alternative for the project

For these main reasons:

- It removes most of the costs and risks of the training environment without removing the benefits associated with it
- It can cope with data anonymization, and UI localization, without requiring further captures on multiple environments
- It is highly maintainable (handling of information vs handling of images)
- It is even more maintainable (one single source for multiple output, multi-lingual, multi-cultural, multi-mode)
- It has a cost of development fairly comparable to the one of software simulations, but compares extremely favourably in terms of maintainability of content, lowering greatly the overall cost of production and ownership.

End User Performance Support: Instant Proficiency

Instant in-application performance support

What if it were possible to generate good quality performance support content directly and automatically from clones and other home made systems such as wikis, short demos, etc? What if, for any type of application, this content, and more to be created especially for performance support purpose, was made automatically context sensitive at screen level, object level, and even aware of end user's likelihood to need it at any point in the application?

Assima's in-application performance support pushes content to end users based on roles and applications. End users can add more content for their own sake, and SMEs or application owners can make campaigns on the spot to push any type of relevant information (communication, performance support, training, etc…) directly to the end user.

By having a system that supports them 24/7 on a timely manner, end users feel more confident in the application, and start contributing into the support system. Knowledge is retained for use by peers. The system is better used. The company's performance is improved.

The Necessity of having a Closed Loop

Measuring ROI

It is all good to deliver the best training environment (the clone), the best performance support (ePSS), but how do you measure the impact of these efforts, and how do you know where to focus the next efforts in order to improve the business further? Put it simply: you cannot improve what you cannot measure.

Measuring key performance indicators on how the actual system is used is paramount. Ideally, the measure shall concern system utilization, performance, compliance, performance support usage, and a set of goals and alerts. With a feed of continuous live data, patterns will emerge and it becomes easier to pinpoint the actual areas of improvements, including new areas of training, areas requiring more performance support, or communication campaigns, or even application redesign.

This feedback loop is the key element to improve even further the business performance, far beyond simple training programs, or performance support databanks.

When 1+1=3

By having integrated Training and Performance Support systems, more benefits can be provided, than if those systems where isolated. Here are a few examples:

- A good part of the Performance Support content can be generated automatically as the best practise defined for user training are the same for user performance. Here is what can be generated automatically:
    - Live Guidance highlighting step by step best practice in the target application
    - Glossary and concept definitions are automatically added into contextual eNotes
    - Macros for repetitive tasks can be generated from best practices
- A user is asking help in some part of the application. The ePSS can automatically check in the training database if this particular user is already certified for this particular transaction of the application. If not, certain action could be taken (display a message encouraging enrolling in a course, block the transaction, etc…)
- Tracking high volume of help calls at the same place on certain transactions can prove end user lack of confidence, and either need for retraining (low retention, or a training that did not focus on learning by doing / by making mistakes), or need for better application design.

Glossary

Term	Definition
Electronic Performance Support System	An integrated electronic environment that is available to and easily accessible by each employee and is structured to provide immediate, individualized on-line access to the full range of information, software, guidance, advice and assistance, data, images, tools, and assessment and monitoring systems to permit job performance with minimal support and intervention by others.
ePSS	Electronic Performance Support System
Ideal training environment	An environment specifically for training that gives the end user access to the real application on a training database, access to a personal coach / trainer and the opportunity for lots of practise and coaching
Retention Rate	How well the end user retains the knowledge transferred to them
Sandbox	A replacement for the training environment where a clone is used in place of the live system / training database.
Software Clone	A true simulation (in the sense of the flight simulator), where the application is replicated using interactive objects rather than screen copies with an instruction design on top.
Software Simulation	A software simulation is generally a pale reproduction of the real application, being a mere succession of screen copies which can be clicked upon in specific designated areas, or hot-spots. For this reason, the name "software simulation" is very misleading and led people to jump to conclusions and expectations way out of its reach and capabilities.
Synopsis	Also called 'Live Guidance', this is a step-by-step guide through a system task that indicates the best practice for the task.

Assima creates technology solutions to support large scale application deployments, delivering measurable return on investment through increased user performance.

Assima's award winning software, training and change management solutions drive adoption, utilisation and organisational proficiency for all your business critical IT change projects.

Assima PLC
CityPoint, 1 Ropemaker St
London EC2Y 9HT
United Kingdom
+44 (0)207 153 1400

www.assima.net

Assima DACG UK
+44 (0)207 153 1400

Assima US
+1 800 608 5373

Assima Canada
+1 514 282 6678

Assima France
+33 (0)1 47 30 71 30

Assima DACG Germany
+49 (0)211 52391 185

Assima Switzerland
+41 (0)21 612 03 79

IMS Assima Denmark
+45 (0)5128 0000

Olas^{IT} Ireland
+353 (0)1 2790 020

Assima Italy
+39 (0)2 67 07 41 05

Assima Spain
+34 91 572 6589