EATM
EUROPEAN AIR
TRAFFIC MANAGEMENT

HUMAN FACTORS

Best practices for e-learning developers in ATM

EUROCONTROL

Edition1.0
Edition date: 08.08.2007
Reference nr: 07/08/07-41

EUROCONTROL

EUROPEAN ORGANISATION
FOR THE SAFETY OF AIR NAVIGATION

EUROCONTROL

Best practices for e-learning developers in ATM

Edition Number	**:**	**1.0**
Edition Date	**:**	**08/08/2007**
Status	**:**	**Released Issue**
Intended for	**:**	**General Public**

EUROPEAN AIR TRAFFIC MANAGEMENT PROGRAMME

DOCUMENT CHARACTERISTICS

TITLE		
Best practices for e-learning developers in ATM		
	Publications Reference:	07/08/07-41
	ISBN Number:	978-2-87497-001-6
Document Identifier	**Edition Number:**	1.0
E-Learning_Practices_V1_Aug07_REP_HUM	**Edition Date:**	08.08.2007

Abstract

This document aims to provide best practice items in e-learning development. It aims to highlight the most important factors that a number of e-learning developers in European ATM felt should be kept in mind to improve efficiency, effectiveness, quality and cross-organisational e-learning development.

The main topics covered are instructional design, human factors (meaning factors relating to behaviour and to other human aspects that need to be taken into account by an e-learning developer to successfully fulfil his/her role), project management and technical aspects.

It is targeted at those who will take on the role of developer within the production of e-learning material in an ATM environment. Others who are interested in knowing more about how an e-learning module is developed and what the main points are that need to be taken into account during the development process will also find parts of this document very useful.

Keywords

E-learning	Computer Based Training	Instructional Design	Online Training
E-learning development	Content development		

Contact Person(s)	Tel	Unit
Max Bezzina	+352 436061 925	IANS/TD/TDH

STATUS, AUDIENCE AND ACCESSIBILITY					
Status		**Intended for**		**Accessible via**	
Working Draft	☐	General Public	☑	Intranet	☐
Draft	☐	EATMP Stakeholders	☐	Extranet	☐
Proposed Issue	☐	Restricted Audience	☐	Internet (www.eurocontrol.int)	☑
Released Issue	☑	*Electronic copies of this document can be downloaded from* http://www.eurocontrol.int/humanfactors/public/site_preferences/display_library_list_public.html			

DOCUMENT APPROVAL

The following table identifies all management authorities which have successively approved the present issue of this document.

AUTHORITY	NAME AND SIGNATURE	DATE
Project Leader	M. BEZZINA	8 August 2007
Chairman HRT Training Focus Group (TFG)	*for* M. PISTRE	8 August 2007
Manager SENSE Programme Chairman Human Performance Focus Group (HPFG)	M. BARBARINO	10.08.2007
Chairman EATM Human Resources Team (HRT)	A. SKONIEZKI	16.8.2007
Director ATM Programmes (DAP)	G. KERKHOFS	20.8.2007

DOCUMENT CHANGE RECORD

The following table records the history of successive editions of the present document.

EDITION NUMBER	EDITION DATE	PUBLICATIONS REFERENCE	REASON FOR CHANGE	PAGES AFFECTED
0.95	05.07.2007	N/A	Proposed Issue for HRT 27	All
1.0	08.08.2007	07/08/07-41	Released issue. Approved at HRT 27 March 2007 (document configuration)	All

Publications
EUROCONTROL Headquarters
96 Rue de la Fusée
B-1130 BRUSSELS

Tel: +32 (0)2 729 4715
Fax: +32 (0)2 729 5149
E-mail: publications@eurocontrol.int

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Overview of the best practices document

Welcome to the best practices document for e-learning developers in European ATM. This document contains a number of ideas for best practices, and suggestions and advice relating to the development of e-learning modules.

The scope of this document is to describe a number of factors which are important when developing e-learning modules. This document does not relate directly to content and pedagogy. However, it aims to highlight the most important factors that a number of e-learning developers in European ATM felt should bear in mind in order to improve:

- efficiency,
- effectiveness,
- quality and
- cross-organisational e-learning development.

It is divided into four main chapters; instructional design, human factors [meaning factors relating to behaviour and to other human aspects that need to be taken into account by an e-learning developer to successfully fulfil his/her role], project management and technical aspects.

The document is organised in three layers each more specific than the one before it. The first layer is the overview of each chapter which highlights the main elements which should be kept in mind. The second layer is the detail found in each chapter. Generally, the principles and good practices are developed in the second layer. The third layer contains further detail and examples which highlight the principles and good practices. This third layer can be found in the appendices to this document.

1.2 Who is it aimed at?

This document is mainly aimed at those who will take on the role of developer within the production of e-learning material in an ATM environment.
Developers have as their main task that of developing the content using computer applications. In practice, however, the same person might also be involved in making suggestions on the scripts and providing ideas so that the learning objectives are appropriately conveyed, and/or managing the development project.

Others who are interested in knowing more about how an e-learning module is developed and what the main points are that need to be taken into account during the development process will also find parts of this document very useful.

1.3 How were these best practices compiled?

The information is mainly based on the knowledge and experience of eight people involved in the different phases of the development of e-learning material in European ATM training institutions [The E-learning Developers Task Force – EDTF].

The work began with a brainstorming session from which the subjects to be presented in a best practices document for an e-learning developer in European ATM were determined.

Following this session, a collaborative document-writing activity was begun, with members contributing and commenting on the work of others. The end result is what you will read in the coming pages!

1.4 Acknowledgements

Before continuing the report, it is essential to thank the following people, without whom the development of this best-practices document would not have been possible:

- Ada Konstantinidou, IANS, EUROCONTROL
- Ángel García Ferrero, SENASA, Spain
- Alexander Ledochowski, Austro Control, Austria
- Helena Ramos, NavPortugal, Portugal
- Jean-Jacques Patacchini, ENAC, France
- Klaus Fischer, DFS, Germany
- Stephen Marshall, IANS, EUROCONTROL

2. INSTRUCTIONAL DESIGN

2.1 Overview

The starting point of this chapter is to provide an answer to the key question, namely -"what is e-learning?" The question of when to use it logically follows within the chapter. In response, a suggested definition is given, along with a table suggesting which e-learning techniques are best suited to various types of learning tasks.

The design process involves responding to three basic questions: "who is the training for?", "what has to be covered?", and "how best to do this?"

The first question does not deal so much with the objectives of a group of trainees to qualify as air traffic controllers, but it deals more with the internal processes and mechanisms by which such trainees learn. Different people have different learning styles, but is one style more common amongst the type of people who seek to get involved in ATM? The answer to this will come from an objective analysis of your students, which will rely on your own experience and that of your colleagues. Some examples of learning styles are given in this chapter.

Next it will be necessary to draw up a typical learner profile. This involves mapping learner characteristics, such as cognitive characteristics, thinking processes, and responses to the learning environment onto the different learning styles. When this has been done within an e-learning design process, it should be possible to design an appropriate learning solution by mapping your syllabus onto the learner profile.

In reality, it is impossible for one style to always suit all, and the advice given is to vary the techniques used, although the training may have a bias towards certain types of exercises and activities.

The second theme of this chapter deals with e-learning course design. Two components stand out. Firstly, there are basic structural components such as the 'look and feel' and the navigation system, which act as a framework in which the content can be housed and interacted with. Effectively designing the tools required to manipulate the software - whose messages may not be overtly recognised by the user but should be very quickly internalised - allows the user to concentrate more effectively on the learning.

Secondly, the knowledge must be integrated into the learning solution. This involves issues such as the design of learning tasks and interactivity, which in turn should reflect the adopted learner profile.

Other issues covered here include content design techniques, the use of games as a learning vehicle, when to use formative assessments, and some important do's and don'ts for adult learning methodology.

2.2 When should e-learning be used?

The instructional design chapter will start by asking questions regarding e-learning and to its use. This will pave the way to later considerations in this chapter on the learner, on the design process, navigational and look-and-feel considerations, and finally, on the methods and tools.

2.3 What is e-learning? – A definition

E-learning can be understood as anything delivered, enabled, or mediated by electronic technology for the explicit purpose of learning[1]. The best way of analysing this question is to first see what the advantages and disadvantages are of using e-learning.

Before proposing advantages and disadvantages to e-learning, we need to see which types of methods and media we can encompass under the umbrella of e-learning if we take the definition above as the starting point and add that it should be delivered through a network [either local, wide or world-wide]. A list of the media used would include: computer-based training [CBT], text-based chats, podcasts, virtual classrooms, email, forums, games, blogs, online downloadable documents, etc. The methods include: interactive self-study, viewing of video material, listening to recorded audio material, reading, exploration, practice, facilitated discussion, etc.

The most 'classical' form of e-learning in European ATM training nowadays can be said to be interactive self-study [CBT available online] but it is important to keep in mind that other online methods exist and try to use them when appropriate.

2.4 When to use e-learning (and when not to)

The following situations have traditionally been cited as seeming to be appropriate for e-learning:

- the target community is geographically widely dispersed;
- the training organisation seeks to ensure consistency in what is taught to this widely dispersed target group;
- a large number of people need to undergo the same training in a time frame that would make this impossible to manage in a classroom;
- the learners find it hard to arrange regular periods of study owing to pressures of work, which in turn makes it hard to convene a group at regular times;
- situations where it is desired that students arrive at a training academy already having attained a minimum level;
- costs and work commitments make it hard for an organisation to release staff for several days at a time;
- the syllabus is fairly stable and will not require constant revision;
- the training organisation is overworked and seeks to transfer some of the study to self-tuition.

[1] Source: TAP Trainer: www.taptrainer.com

The drawbacks include:

- possible feelings of isolation and alienation can develop in the learners;
- the learner can feel isolated and unsupported when technical problems occur;
- the fact that some people respond more readily than others to the computer as a learning tool;
- interactive or hands-on e-learning material is very expensive to produce;
- there is not enough easily accessible training material out there;
- sometimes human instructors can feel threatened by a technology they fear might replace them;
- sometimes human instructors feel threatened by the fact that they might not be able to adapt to the new methods.
- sometimes human instructors are concerned that the introduction of e-learning is a backdoor method with which to increase their workload;
- it can be see as learning on the cheap, and some feel that it benefits management more than instructors and students.

The comments above are not listed in order of importance, they are merely the expression of themes that regularly occur when discussing e-learning.

2.5 How is e-learning best used?

In the early to mid-1990s, with the dawn of the WWW and with the optimism that prevailed about all Internet-based technologies, quite a few people argued that e-learning would eventually substitute other more ‘classical’ forms of training.

A dot-com bubble-burst, a number of failed projects and thousands of hours of e-learning experience later, very few are left who would share this viewpoint. It has been demonstrated that e-learning is a powerful learning method that should be used when appropriate.

The focus and the emphasis should be on the learning that needs to take place and not on the methods and media used. The latter should be there only to facilitate the learning [like a referee in a sports game]. The current training approach is to blend methods and media to make learning as effective as possible while enhancing efficiency. The skill is in finding the right ingredients [the blend] which would then create the best learning effect whilst taking into consideration the constraints imposed by the operating environment. E-learning is seen therefore as a way of complementing other methods within a training curriculum.

2.6 Getting to know the learner

After having analysed what e-learning could be, the next part of this chapter sets out to identify the end-user of the product; the learner (‘who’) and the task (‘what’) which needs to be taught. This will help us as e-learning developers to adapt our training solutions to the needs and preferences of the ‘who’ and the ‘what’.

2.7 The learner and the task

'De gustibus non est disputandum' is a Latin proverb that translates as 'There is no accounting for taste' meaning that everyone's tastes are different. Not only tastes are different but also characteristics and styles. This also applies to learning.

Getting to know the learner's predominating learning styles and characteristics will allow us to adapt the solution to these. In some areas, for example ATC, particular styles and approaches to match these might prevail. In other situations, where styles might be more varied, a blended approach would really then be the solution to make the training as effective as possible.

2.7.1 The learner profile

Purpose

How do people learn and why is this important? Let's answer the second part first.
It is important for a learner to study in a manner which blends in with their 'preferences, idiosyncrasies[2] and ways' otherwise, lack of interest, frustration and alienation will probably prevent them from learning effectively. As such, the learner profile is a holistic concept. It relates both to the learner and to the training organisation, given that it presumably wishes its students to learn as efficiently and effectively as possible.

So how do people learn?

2.7.2 Learning styles

A study to determine whether different learning styles existed presented students with a survey asking them how they would respond to the task of setting up their voice mail on a newly installed phone system. This investigated the use, if any, which they would make of the manual and other colleagues. Would they dive in and wrestle with the task? Would they at any stage call a halt and look for some sort of printed or human support? Would they acknowledge the 'font of knowledge'? Perhaps they would prefer to meekly wait until their office colleagues had solved the problem and then, by whatever means necessary, get them to do the task for them?

From experiments like this, it is possible to identify some typical common learning styles. Let's categorise them as follows:

Directed – this type of learner is motivated but seeks guidance and confirmation from the font of knowledge. They will read the manual and probably successfully set up their phone. If they can't, then after making a respectable effort, they will get someone to help.

Investigative and competitive – this type will dive in without reading the manual, and might refuse to consult it at any stage. If they are successful, they will 'really work out' the system for themselves, and they will 'know' it, so that later they will be able to solve the problem again, and maybe other problems with related aspects.

[2] Definition of idiosyncrasy: a structural or behavioural characteristic peculiar to an individual or group

However, they might miss something, as they are reluctant to get confirmation (do they see this as a sign of weakness?), although they will probably be convinced that they are right. This type of learner might be very happy to share information (which might be very good) since this broadcasts their success, on the other hand they might withhold the information from people who are ahead of them or threatening them in the race.

Investigative and cooperative - this type of learner will dive in without reading the manual as the task is seen as an interesting challenge, but they will not keep on battering away at the point of strongest resistance and will involve a colleague or the manual when they get stuck. They get some satisfaction from solving the problem, but really just want to make sure that friends and colleagues can leave messages. Could this type of learner be suited to group-based learning activities? Such a learner would be more than happy to pass on their knowledge, but might not regard it as a high priority task.

Dependent and reluctant – this type of learner will either play helpless and wait for someone to take pity and resolve their problem for them, or alternatively, they might pretend that they don't care and are refusing to set up the system precisely because they don't want to be bothered by people leaving messages.

They may show little initiative in taking the lead in their own learning, either perhaps because they are reluctant to learn – they may be stubborn, reluctant to be 'coerced[3]', or just plain lazy - or because they lack the confidence to do so. They are likely to need a lot of structure and encouragement. This type of learner, either inside the classroom or on-line, needs to have the structure provided to them otherwise they might run a serious risk of getting lost or demotivated.

You may well disagree with this analysis, which is only conjecture and not based on empirical research, but please at least go on to make your own. Also ask yourself, what kind of learning activity might be suited to such views of the learner profile?

It is important to be aware of the preferred learning styles of the end-users when designing an e-learning module as this will help in designing training and using methods to which they best respond.

… and learner characteristics

Learning preferences can be divided into characteristics. Within some of the characteristics, each individual will have a preference. In other characteristics, each individual would be strong, medium or weak. Learner characteristics are a combination of:

A. cognitive characteristics, including the learner's preferred learning channel and thinking style;

B. metacognitive characteristics, meaning how a learner thinks about their own thinking processes. This would include the ability of the learner to plan, monitor and adapt, and self evaluate a learning experience;

C. physical characteristics, such as environmental phenomena that might distract the learner and reduce the effectiveness of the learning, and also;

[3] Definition of coerce: to force someone to act or think in a certain way by use of pressure, threats, or intimidation; compel

D. affective characteristics, including the learner's self-confidence and belief in their own capability to organise and execute actions necessary to learn. The presence or lack of self-confidence strongly influences the motivation to learn.

A. An important ***cognitive characteristic*** is the learning channel. People can learn though visual, auditory or tactual modes or any combinations of these. However, different users will have different preferences; if visual learners prefer a visual presentation, audio learners might prefer discussions or lectures. The question then emerges: "how to satisfy them all?" Within the design of an e-learning module, it is important to keep in mind this aspect and try, where applicable, to present things visually and aurally but also to include exercises which focus on learning by doing.

Another cognitive characteristic relates to the theory that people have different thinking styles. The distinction here is between field-dependent and field-independent thinkers.

Field dependent thinkers rely on external referents and on external structures to learn, while independent thinkers rely on themselves and impose their own structure to the learning experience.

B. ***Metacognitive processes*** deal with how a person thinks about their own thinking processes! How they decide on a particular strategy, what goals they draw up, how they organise the task, and how they gauge their success. Also, metacognitive processes allow the learner to monitor themselves and their progress and adapt during training as and when needed. At the end of the training, metacognitive processes allow the learner to self-evaluate their success and prepare for future learning experiences. The presence of metacognitive processes varies in different learners and ranges from strong to weak.

C. ***Physical characteristic*** are those which could distract the learner during training. Some are distracted more than others from a particular external distraction. Sound and visual noise are two examples of external distracters.

D. Finally, ***affective characteristics*** relate to how people might feel about their learning environment. Who thinks they can sail through the course without problems, who fears they might look bad, who has already been judged as a slow learner, and contrast those who find an internal motivation to study with those who are studying because it cannot be avoided. Perhaps the motivated are motivated because they have worked out a successful approach to learning?

It is important to be aware of the existence of these characteristics during design as it helps the developer keep in mind the particular characteristics of the learners when planning how to make the training more effective.
In an environment where there is a lot of variety in the characteristics of those who need to learn, then the best approach would be to be flexible and adapt to individual or group needs.
In a varied environment, in terms of developing e-learning, it is important that a training module does not overly concentrate on one method in particular, but is as 'democratic' as possible.

2.7.3 Appropriate learning solutions - the learning vehicle

Given the above, what might make a successful model? It would seem that a variety of approaches gives the different types of learner at least some tasks to which they are suited, or at least avoids a situation in which a talented person might be driven away by training that seems at every step to be unsuited to them. On the other hand, it should also be noted that such variety might actually be problematic for some people, because they feel lost with all the options given! As stated above, variety should be coupled with flexibility in the delivery approach so as to allow those delivering the training to adapt the strategy to the individual or group's needs.

Additionally, if you feel that a preference generally predominates, for example, in the ATC world whereby those working in this domain generally enjoy approaching a task in a particular way, then this will give you a strong clue as to the types of learning activity that might be most suitable to the community of controllers as a whole.

Bearing in mind that no one has the budget to offer alternative methods of study for every component of their training course; this tends to suggest that the perfect training course is impossible to design. In these circumstances, it does seem logical to suggest that it is best to use a variety of approaches to e-learning and to avoid solutions that become totally divorced from more human, classroom-based instruction [i.e. favour a flexible, blended approach].

If you would like to read more on learner profiling, here is a link which we found useful: http://www.netg.com/pdf/NETg_WhitePaper_Profiling_the_Adult_Learner.pdf

2.7.4 The learner is also an adult

Apart from the theories on learning profiles and styles, there is another dimension that should be considered when designing training. This is how adults learn. The adult learning theory is also very vast. Below you will find a number of highlights which in a best-practice scenario are taken into account by the development team when designing e-learning material.

2.7.5 Considerations on the adult learner

Preamble

Through the research we have done while preparing this document we found that there is no general theory of adult education because of the incredible numbers of variables (Brookfield). Each adult is different.
Cognitive styles, learning styles, physiology, culture, and personality are some of the variables. But if you are prepared to listen to learners, they may teach you what you need to know…

Following you will find a summary of our research, especially relating to instructional design.

Avoid

1. Avoid meaningless learning.

Be careful

1. Be extra careful when dealing with learning which re-assesses old knowledge [as you might encounter strong reactions from the adult learner].

Do

1. Connect new learning to existing knowledge.
2. Organise training around problems participants will probably have experienced rather than around subjects.
3. Acknowledge that different perspectives to an issue exist [as different people see things from their own angle].
4. Keep in mind whether your students are normally learners who need structure or independent learners who just need little guidance and can learn mostly alone. Adapt your design accordingly.
5. If the course includes activities and assignments, provide deadlines but retain flexibility when determining them as learning is optimised when adults have a say in deciding their own learning pace.
6. Give the adult learner as much control as possible. [Try to design discussion groups, etc. in the training].
7. Include activities in the training, as learning by doing is very powerful in adult learning.
8. Consider the motivation factor in adults[4].
9. Include communication mechanisms within the training. These may take the following forms:
    a) Discussion (via email or in forums),
    b) Argumentation (making a case for a particular viewpoint),
    c) Inquiry (student-constructed response to an instructor-posed question).

Facts [take these into account when developing training]

1. Adults pay more attention to detail.
2. Adults tend to be problem-centred.
3. Adult learners bring with them different levels of ability to think critically, analyse results, etc.
4. Active learning and collaboration between learners get the best results in adult learning.
5. Group work has a high benefit on adult learning [whenever possible include group-work activities in your courses].

If you would like to read more about adult learning, we found this site particularly interesting: http://roberta.tripod.com/adulted/methods.htm

[4] See EATM Guidelines for using e-learning document

Caveat

We believe that the theory above needs to be viewed in pragmatic terms; the information presented is very relevant and can be beneficial during the design process, however it should not be generalised, and the user should always think of whether and how it applies to her/his local context.

Having considered the learner, another factor that needs to be considered is the subject to be taught. The combination of knowing the 'who' and the 'what' will enable us, the e-learning developers in ATM, to design training which is effective and efficient.

2.7.6 Considering the task: a focus on ATC

The world of ATM is a very complex environment and it can be very hard to understand technical elements without relating them directly to their application.

ATC is a task-driven activity, meaning that the knowledge and skills air traffic controllers [ATCOs] need to develop are based on activities. Many studies have been carried out on these activities[5]. These have shown that to become a competent ATCO, various knowledge and skills are required.

The main tasks of ATC are related to safety, capacity, assistance, etc. To provide these services, the controller will need to:

- solve complex problems;
- elaborate strategies;
- apply actions plans.

These actions plans are the end-result of various cognitive processes.

In the framework of learning these processes, we need to define:

1. Static knowledge
   e.g. air law, navigation, aircraft performance, geography, etc.
2. Dynamic knowledge which is the result of mental processes
   e.g. "My analysis of the situation leads me to the conclusion that these two aircraft are in conflict";
3. Action plans related to traffic situations
   e.g. Radar vectoring to solve converging traffic;
4. "Quick solutions" which are complex mental processes formed through experience.

From the above framework, we can therefore talk about knowledge organised in multi-layers within ATC.

[5] For example: EUROCONTROL, EATM: Integrated Task and Job Analysis of Air Traffic Controllers Parts 1-3

2.7.7 Where would e-learning fit in?

Current e-learning methods are not efficient to practice “quick solutions”, i.e. complex mental processes. As a consequence, these should be trained and developed elsewhere.

However, CBT or other e-learning material could be available to develop elements 2 and 3 in the framework above, especially if the training is based on realistic traffic situations or adapted games.

For type 1 knowledge, e-learning can be a good training method. Nevertheless, a sensitive issue would be the way the subject is presented to the ATCO [or student ATCO] and could determine the effectiveness of the method.

The presentation of framework item 1 on e-learning could be based on different options.

Example:

- text: literal presentation of the assumptions
- graphics: symbolic or realistic
- operational situation; interaction

Taking a practical example from initial ATCO training will help illustrate the approach:

- Subject : aircraft performance
- Presentation:
  - Text : “in case of constant indicated air speed, aircraft true air speed varies with altitude”
  - Graphics: “curve representing the air profile”
  - Operational situation: “set of radar pictures showing the evolution of ground/true air speed with no wind”

For the student ATCO, each of these representations can be understood. However, in each case he/she will ask the question “what does this mean for my job? Where can I apply this knowledge?” So, every time the knowledge can be illustrated or related to tasks, the training will be more efficient because the ATCO not only will remember the assumption (“what?”) but also the explanation (“why?”) and the possible effect (“what do I do?”).

If the ATCO task analysis has been done, it can give information on:

- “What do I need to know in this kind of situation?”
- “Where is this type of knowledge needed?”

In conclusion, for task-driven jobs like ATC and flying aircraft, it is more efficient if the knowledge can be related to or introduced by elements of identified tasks. The operator can then relate this knowledge either to comprehensive actions and constraints or to already-encountered situations.

2.7.8 Conclusion: The learner and the task

In this first part of the chapter on instructional design, we considered the nature of our end-user and the nature of the task to be taught. These two factors will need to be taken into consideration during the early stages of instructional design.

Having described these aspects, the next part of this chapter will have a closer look at the instructional design process.

2.8 The design process

The design process described in these guidelines is divided into two parts. The first part defines the process through which the learners, the domain and the training will be approached. The second part describes how 'crude' subject matter is translated into interactive content.

2.8.1 Part 1: Who – what – how

Within this part, we will look at the process which can be used to design e-learning material. The process will build on the identification of the learner and the task.

The first step within the process is to identify and precisely define the target audience. The process starts from the target audience and follows by defining all elements needed. This will provide the milestones and competences required to reach the goal.

The pedagogical strategy should result from all these factors and drive the design of the future product.

In order to identify all the details you should ask yourself:

Who is the learner?

- What kind of population will be following your e-learning programmes?
- (Keep in mind that a target audience is not necessarily homogeneous [e.g. age, computer literacy, etc.])
- What does the learner know before starting the learning session?
- What is their way of thinking?
- What is in it for them?

- What kind of competences in using computers are requested to be able to follow an efficient e-learning session?
- Is the learner used to this kind of technology?
- Is the e-learning session designed to answer immediate questions or it is done to improve long-term knowledge?
- The dimension

  - How many learners are we trying to reach?

What is the domain to be taught?

The domain to be taught can strongly modify the design of the e-learning project and the technologies needed. For example:

- If the domain is quite technical and will need a lot of either definitions or external references, the learners will need a structure which acts like a dictionary or lexicon so as to facilitate reference.
- If the domain is quite theoretical and wide-ranging, the programme could be divided into small elements that can be learned or skipped according to the instructional method used.
- If the domain includes practice on advanced mental processes or skills like the ones used in ATC (e.g. problem solving), representation of the situation needs to be quite realistic [as a consequence, specialised PC performances in terms of graphics and processor capabilities could be needed]. Also, a link with other methods, such as simulation, should be made.

Another element which needs to be taken into account is the relationship between the domain and the learner profile. The ergonomics, the terms used and even the way of learning could be slightly different according to the learner profile, especially in the world of ATM.

Also, within the design, different levels of acquisition may be allowed. For example to allow progress:

- either from novice to expert; or
- from general to specific.

These different options can be linked to standard requirements like for instance Common Core Content, where objectives are described with levels of taxonomy which lead to a certain kind of learning.

2.8.2 Part 2: From subject matter to interactive matter

The second part of the process, after the who, what and how have been identified, is to translate the ‘crude’ subject matter to interactive matter in an e-learning module. For this, the documents and the knowledge which constitute the subject matter on which the future e-learning module will be based need to undergo a series of adaptations so that they can be incorporated into an on-line (and very often multi-media) environment.

For this to happen, a number of roles need to interact together within the development project. Each role takes care of their area of expertise. Below is a description of the roles that need to be present in an e-learning development project:

- The developer: the entity and person(s) in charge of developing the module in a computer language. This role is involved in design, selection of media and coding.
- The script-writer: the person in charge of writing the script. He/she needs to consult with both the SME and the developer and needs to take their needs and constraints into consideration.

- The script-writer also needs to be familiar with the requirements of both the other parties in order to clarify possible communication problems between the SMEs and the developers[6] .
- The subject matter expert [SME]: the person or people in charge of providing the subject matter for the development of the module.

It is also very important for the project team members to understand that, even though at times the same person acts in different roles, these roles are very distinct and require a specific set of skills and background knowledge.

To give a brief example, the SME should not be expected to single-handedly encode an interactive multi-media module in a computer programming language, a developer should not be expected to single-handedly extract the 'crude' subject matter information and convert it to an e-learning module.

A way to bridge the gap needs to be found and we suggest: discussion, via the role of the script-writer and through scriptwriting.

The script-writer is the one who will make the link between the subject-matter requirements and the development possibilities. The person fulfilling this role needs to understand the requirements and constraints from both sides and thereafter needs to bridge this gap.

It is suggested that he/she does this by writing a detailed script[7] which divides the future e-learning module into a number of pages and within each page there is a detailed description of what is to be expected. These expectations need to take into account both the learning objectives and the subject matter which needs to appear in the module, as well as the design which will be used (e.g. interactive exercises, pictures, sound, movies, page setup, etc.).

In this way, the SME can see what to expect once the module is developed, and the developer can understand the requirements and focus on the creativity side of putting concepts, ideas and text into a multi-media format.
The project manager has the responsibility of seeing that the script proposed is agreed by the project team members. The project manager shall also have at least basic fundamental knowledge on ATM education (instructional design, knowledge on how different ATM personnel react to different teaching methods, etc.).

[6] Note that one of the members of the project team needs to take the role of an instructional designer. Often this is shared between the script writer and the developer.

[7] An extract of a script which has been produced in the development of an e-learning module is available in appendix 4.

In this way, all those involved are responsible for tasks in which they are competent. The expectations of the parties involved should also be agreed at a very early stage. (This should reduce the expectation gap at the end of the development.)

The project team should function in such a way that everyone builds on the skills, knowledge and input of the others. In this way, while everyone sticks to their allotted role, pro-active involvement (creative ideas and problem-solving for example) from all team members will improve the end-product.

The following is a table which describes the specialised tasks which need to be carried out during the development phase, and who should do them:

Developing E-learning modules **Activities:**	**Project Manager**	**S M E**	**Developer**	**Script-Writer**
Managing the development phase	R			
Extracting training objectives & drafting a storyline.	I	R	C	C
Scriptwriting	I	C	C	R
Providing subject expertise		R		
Providing Pedagogical expertise				R
Providing multi-media expertise			R	
Developing the material on-line	I	I	R	C

R = responsible for this task, C = contributes, I = informed

2.8.3 Conclusion: design process

In this part we have seen that during the design of an e-learning module, an e-learning developer within a project team needs to start by analysing who the training is aimed at, what needs to be taught and finally, how it could be taught.
Once this part of the process is done, the next stage is to convert the subject matter into interactive material. This process involves three main roles, required to work together. These are: the SME, the script-writer and the developer.

The project management part of the development of an e-learning module will be described in the next chapter: Project management.

In the next section we will consider factors which directly concern the developer. These are the navigation and look-and-feel elements of an e-learning module.

2.9 Navigation and "Look and Feel"

2.9.1 Keep it simple!

The most important thing to bear in mind is keep everything simple. We should not confuse the user with complicated navigation and an over-charged look and feel. The navigation and look and feel are only the facilitators that lead to content.
The focus is on the content.

2.9.2 Navigation principles in an e-learning module

In order to offer the user a complete e-learning experience we should take into consideration the following generic characteristics of adult learners:

1. They have limited time to devote to the module due to other obligations.
2. They prefer learning by doing.
3. They need to have various choices so that they can control the way they study.
4. They have specific expectations of the module they are studying.

In order to satisfy the above characteristics, we need to think in terms of:

- ***a "multilayered" navigation***: This means that users can reach their "target" topic following different paths.
  For example, the interface should have arrows to move back and forth and at the same time there should be a menu, or a list that allows direct access to the topics.

  Having multilayered navigation is important because research shows that website users do not form mental models of a site. Therefore, the integration of numerous navigational links is very important (Spool, et al.).
- ***a "second-level interaction":*** This term specifies all the "additional" types of interaction offered to the student through the basic interface. These are necessary to allow the user to interact more actively with the course. For example, keeping digital notes, the search function, glossary, and download material related to the subject and to the learner's work.
- ***status indication***: The learner should know at anytime what their status within the module is. This means that the user can easily get to know the percentage of the course they have completed, and the remaining content.
- ***course map***: Unless it is a guided course, the learner needs to have direct access to any topic of the course. If it is a guided course the learner should at least be able to see the whole course structure.

- ***create their own learning track after a pre-test***: Whenever technically possible, in order to facilitate learning and to motivate the end-user, a pre-test could determine each student's level of knowledge and then dynamically customise the content by showing only the topics that the learner will need to study (personal learning track). The rest of the topics should always be available to the learner but as an alternative option.

The above principles have been added with the intention to provide tips on what could be done to facilitate the user's navigation within an e-learning module. These principles would need to be adapted to the context and training needs.

2.9.3 Basic principles on look and feel in an e-learning module

The basic look and feel of an e-learning module is very important to the learner and to the effectiveness of the training. It can make the difference between attracting them to follow and complete the module or demotivating them from continuing
Below there are a few good-practice tips in this respect that could be taken into account.

Before continuing however, we would like to note that the look-and-feel part of the subject is related directly to ergonomics and graphic design. Most e-learning developers are not experts in this field. The most advisable thing, especially when designing a template which would be used for a number of modules, is to consult an expert so that they can professionally advise you on these aspects.

Our main target is the content

We should keep in mind that when we want to offer an effective e-learning experience to our students, we should make their journey into the modules as relaxed as possible. Our main target is the content. The student should not be distracted from it because of a fancy interface. A fancy [tiring] interface, full of sparkling colours and blinking animations may offer the developer the satisfaction of having proven his/her skills, but the result will be negative for the student and for the training objective.

If we want to make something "fancy" we should look at parts of the content that we could enhance, not the interface.

Colours and Design

The colours that we use should be neutral and the design should be limited to serving the interface needs. One rule is not to waste space on irrelevant design since we will need the maximum space for the content.

A second point to check is readability. We should be very careful of the colour combinations that we use.

Colour coding and formatting conventions for the content [e.g. red for objectives, blue for examples, italics for comments, etc.] and using this code consistently during a module is important. The student will get used to the code and this will assist them in looking for items in a more efficient way. Also, the eye is attracted to colours and schemes, so using the appropriate colours to convey certain messages is important.

If you would like to read more about using colours, we found this site particularly interesting: http://www.atpm.com/9.08/design.shtml

Fonts:

Another aspect of the look and feel is the choice of fonts.

Do not use more than 2-3 fonts during a module and try to find fonts that are commonly available by users.

If you would like to read more about using fonts, we found this site particularly interesting: http://www.efuse.com/Design/web_fonts_basics.html

Interactive material

One of the advantages of CBT is that many items can be reproduced in a realistic manner [e.g. by the use of program such as Flash]. Realistic presentation of the interactive material [e.g. if a flight-plan form is to be presented then present it in the most realistic way possible] is a very important element which makes the module credible and enhances the motivation of the learner to continue.

2.9.4 Conclusion: navigation and look and feel

Navigation and look and feel are two very important design elements. Their appropriate use will ensure that the learner focuses on the content and that the content will stick out because of the chosen underlying design.

In the next part we will look at various methods and tools that could be used in on-line training.

2.10 Methods and tools

On-line training, as mentioned earlier, can take many forms. Even in interactive CBTs, various methods and media can be used for training purposes.

This section aims to describe a number of these and suggest when they would be most appropriate. The list below is by no means comprehensive. However, it is the list that the EDTF which compiled this document considered should be mentioned in these guidelines.

2.10.1 Methods and media to be used in e-learning when teaching: skills, data retention and knowledge

- The methods and media used in e-learning to display the contents of a learning activity in a certain medium are essential to maintain student motivation until the end of the learning activity and thus to comply with the training objectives suggested.

- There are methods allowing these contents to be displayed in the distance, by means of a text, an audio file, a video, flash animations, etc.
- Contents in text format are used for reading. In this case, it is very important to take into account that according to different studies it is guaranteed that only 25% of the information displayed on the screen is picked up, whereas information reproduced on paper is picked up at a rate of as much as 70 %. Several factors such as the size of the screen or the font size, colours, light effects [already discussed above], etc. together with the stiffness caused by the computer after being sat in front of it for several hours, may contribute to these results.
- Another good alternative for the display of contents is to use videos or reproduce audio files in which the student uses both ears and eyes to assimilate the contents. It is necessary to use an active approach to making an audio/video edition, combining elements which prevent the student getting bored.
- For an efficient and effective use of video/audio file, it is very important to take into account that technical requirements are not widely available yet. The student should be able to watch the video or hear the audio file at the desired moment without loading it in the local computer. However, to do so the computer must have a network connection with a wide bandwidth in order to obtain the appropriate quality.
- The latest novelty for displaying contents is the use of Flash animation. Flash is a program of vectorial design which, owing to its versatility and to the small size of its files, has brought about a revolution in distance learning.
- This format allows examples and contents to be displayed with animation techniques combining graphics, sounds and videos. This will allow student interaction with fundamental concepts, animated examples, simulations, resolved exercises and studying guidance, which will in turn contribute to a better assimilation of the contents and the skills.
- The selection of one method or another for displaying e-learning contents will depend on the skills to be acquired by students (learn a skill, retain data, gain knowledge, etc.), always taking into account the level of computer literacy. [How at ease will they be in front of a computer?]
- What kind of e-learning application do you really need? Research has found that learning applications fall into four major categories.

The table below describes the categories and also suggests the tools to be used

Category		Example	What the learner will do	Tracking Needed	Tools that could be used
1	***Broadcast of new information***	"Introduction of a new working policy"	Read	None	Text Audio/Video
2	***Important knowledge transfer***	"Introduction of a new procedure which replaces an old one – e.g. the introduction of 8.33 kHz spacing above FL 195"	Read, listen and answer some questions	Who took this? Did they get it?	Text Audio/video Animated examples Auto-evaluations
3	***Developing new skills***	"Learning how to vector"	Read, listen and try out new skills	Did they really learn? What score did they get?	Text Audio/video Animated examples Interactive games & simulations Auto-evaluations
4	***Creating complex associations***	"Integrating a number of skills to provide separation"	Read, listen, try new skills and become competent	Did they pass? Are they certified?	Text Audio/video Animated examples Interactive games & simulations Courseware with assessments or certification exam

2.10.2 Assessments

First of all, it is important to make a distinction between summative[8] and formative assessments.

Formative assessments

Formative assessments allow the end-user to learn from the questions asked and to stop and assess whether there is a match between the perception they have of the learning experience and the answers to the assessment exercise.

[8] Please find definitions for summative and formative assessments in appendix 1.

Because of the nature of asynchronous CBT, having formative assessments embedded within the e-learning programme is very beneficial. These assessments could take different forms, from classical questions but which give feedback depending on the replies provided and pointing to the page, chapter or module where the item was presented, to games, simulations and other original ideas which allow the student to assess the knowledge/skill they have acquired.

There are a number of tips that could be useful when designing formative assessments:

- Formative assessments are first and foremost for the benefit of the students, therefore it is suggested that the results are not tracked.
- Feedback needs to be incorporated within the question being asked.
- An analysis of the results could be incorporated following a set of formative questions.
- The questions should be worded in a challenging way; where possible the question should rephrase the concept and not copy the element word for word from the module.
- In the event of an incorrect response from the student, pointing to the place within the module where the answer to the question is found can be a good practice.

As a conclusion, it is highly recommended to include formative assessments at regular steps within an e-learning programme.

Summative assessments

The main aim of summative assessments is to check whether the student has achieved the learning objective. Creating summative assessments is a job in its own right, as there are elements of:

- creating the right type of questions at the right level to ensure that the assessment faithfully represents the training module learnt;
- ensuring that the assessment really measures the student in terms of the learning objective in question and not on other objectives;
- ensuring the security of the test and the verification that it is the right student who did the assessment;
- etc.

It is therefore beyond the e-learning developer's area of expertise. If an e-learning developer is asked to design a summative assessment, it is highly recommended that this person undergoes specific training on the subject.

Below there are a number of considerations on summative assessments and e-learning that a developer should be aware of, taken from the guidelines for using e-learning in ATM document:

"Learning management systems are usually capable of measuring the students' progress through a module, the time spent in it, as well as results from an assessment which would be incorporated. The use of online assessments may be the logical answer and one of the options.

However, a number of constraints exist with the current mainstream technology available. These constraints are mainly due to the fact that it is easy to cheat on assessments over distance and the verification that the actual candidate responded to the questions is very weak. The motivation of having computer assisted assessments is that it would avoid the resources needed to correct them.
However, this is only valid for closed questions, e.g. multiple choice, true/false, yes/no questions. For open questions [even though some software solution sellers would claim differently] human intervention would still be needed.

Therefore, depending on how serious the assessment is and the nature of the requirements [closed / open questions], delivering via distance e-learning may or may not be an option.

With very serious assessments, in other environments, computer assisted assessments are still held in secured and designated premises where candidates have to present themselves with an official identity card and once admitted are not allowed to speak to anyone.

For less serious assessments, which could serve as an extra tick, together with for example, the time and progress recordings, closed questions could be incorporated within the modules in the form of exit questions. These questions could then act as gateways to further training, meaning that if the student does not pass, he/she would not be able to proceed with the training.

Average learning management systems are reliable enough for content to be delivered through them, however not generally enough for secure, high stake assessments. In order to make the technology as reliable and as secure as possible a lot of investment would need to be made [think of Pareto[9] 80 / 20!]. The cost benefit analysis of this might not look that good. Even in less serious assessments, it needs to be kept in mind that most systems will not be 'bug' free, and cases could still exist where items are not recorded correctly on an LMS. To have a bug free system it would cost a lot of money. [This is one of the reasons why ATC systems cost a lot of money to produce because they need to be reliable over 99.99% of the time…]

The alternative is of having paper summative[10] assessments. At the time of writing, for many aspects it is still the suggested option."

Games

Games can be a very powerful and effective learning tool. Games can be used as metaphors. They engage the learner and are virtually risk-free for students to test their skill or knowledge in a fun and relaxed way. Computers lend themselves very well to gaming and over the last decade educational [or training] games have been on the rise. However, designing a good game is a challenging task. They need to be well thought out and well executed. They normally need a lot of time to plan, design, develop and to test and often this means money. So they really need to be worth it!

In European ATM, playing games to learn is a relatively new concept, however a few examples have shown that games can teach ATM 'stuff' in an alternative and challenging way.

[9] Pareto analysis: http://www.answers.com/topic/pareto-analysis

[10] Please find definitions for summative and formative assessments in appendix 1.

We have seen a game that involves driving a car through a snake-like maze by giving it headings – players win points depending on the time and lose them if they stray from the route! We have also seen 'aliens' that in order to survive need to decode METAR, or aircraft flying around a Pac-man like maze from one reporting point to another whilst avoiding thunderstorms. The player had to input channels while 'flying' in order to get to the next reporting point to pick more points.

Below are a number of considerations compiled by the EDTF on learning games and ATM:

Games are good:

- for teaching codes and symbols – codes and/or symbols encapsulate ideas much more concisely than text. This is why a game could make a visual representation of these codes and symbols more effectively than a text, for example. If you have to teach what codes or symbols mean, consider a game where for example the language spoken is in fact made up of these codes;
- to use as metaphors – if you need to teach something which is of a contentious nature, then use a metaphor and develop a game to avoid learners getting bogged down arguing about the detail of the real world environment. In gaming terms, a metaphor is a game which mirrors the structure of a problem, skill or a function but uses another context. For example giving clear directions to a tourist to find a place in town could be a metaphor to giving clear instructions to aircraft, especially where the point is about the clarity and structure and not about the phraseology.

When designing games:

Games do not necessarily need a high level of graphical sophistication. They do, however, need a high level of sophistication in the concept behind the game.
Games should have a competitive nature: incentives for players are very important - alas, providing rewards is difficult where there is no room to award actual material prizes. This means concentrating on:

- developing competition (player-player or player against their own personal best) - best scores, etc.
- developing intrinsically satisfying features - likable characters, n rich style of graphics and sound, humour, the 'buzz', attractive goals, etc.
- providing temptation that leads the user towards something still more attractive and then denying it! - motivation can be provided by splitting a game into levels so that the player must score well to access the next level. No one wants to be excluded from the next level!

There are two typical relationship structures for games - games against the computer and games against another human opponent in which the computer acts more as the 'board'.

Either type of game can work, but games against another human depend on another human being around, and often on the players sharing a computer. But they can be great for competition and variety. By default most multiplayer games have a basic mode where the player can pitch his skills against the impersonal computer as a fallback.

Games should have a point - for example, if the game is about reinforcing good practice, it should contain a component that irritates the player. During the game, this 'real' point may be hidden, but at the end you just need to make the link between the 'point' and the 'irritation' and the students should get the message.
Time components are great devices for focusing the mind - games played against the clock are good for mimicking a stressful environment as well as increasing the element of competition. They also provide an obvious qualifier for scoring systems.

Some further information about games:

There are many different philosophies underpinning games and here are a few:

- 'Be a god' games - where the player gets to be a god and can organise the world as they see fit. But what of the consequences, and how powerful actually is this god and how quickly can he/she respond when things start going wrong? These kinds of games are particularly suitable if you want the player to understand the complexities of a certain environment by giving them the 'power' to act on the environment and then see the consequences.
- Shoot 'em ups and other similar games are quite good when the skill being trained involves fast thinking and reflexes - but watch you don't shoot the good guys!
- Exploration or maze games, the word maze being used metaphorically - these involve the player collecting tools (gaining knowledge) which allow them to find the key to progressing to the next level, and eventually to escape from the maze or reach a goal.
- Racing games, as mentioned earlier, for example racing to arrive at a destination by giving the right 'headings' to a car.

Two final considerations on games:

Is everything appropriate for a game? Does the importance of the point justify the development time? Does the type of information lend itself to game playing? Can everything, no matter how dull, be abstracted into a suitable format for a game?

Be careful when thinking of designing a game. Remember that the main objective is to achieve effective learning. Don't let the thrill of the possibility that you would be designing a game interfere with a rational analysis of the considerations mentioned in the paragraph above.

And one last word of warning; from experience we found that some people hate games. Be careful that your ideas and suggestions are acceptable to all the other people involved in the project.

Case studies

Case studies can be used when you need to illustrate theory with an example. [Remember that adult learners generally prefer learning by doing!]

In a case study, the student is presented with a situation that has happened [real or made up]. The training tool could ask how they would have handled the situation and later ask the student how the situation was in fact handled; the good things and the things that could be improved.

Unfortunately, most of the time case studies remain fairly theoretical as all the student is being asked to do is to make suggestions and discuss the solutions proposed by others.

Case studies could be built in e-learning modules which are part of a broader training programme where the learners then meet to discuss the case.

As a summary, case studies are a good way of making sure a course is not purely theoretical and providing the opportunity to illustrate the theory by examples.

Blogs, wikis, podcasts, webcasts

The above are four other tools which although not widely used in ATM on-line training, could be considered. Below is a brief description of each:

Blogs

In simple terms, a blog is a web-site which can be easily published via a web-browser. Blogs are mainly used by individuals or groups to write posts [articles] on any subject of their interest on a regular basis. The interest of these blogs is that they can be used as logs [as their name originating from web-logs indicates] and in our case as learning logs. An e-learning course tutor could create a blog for their students and during the course they could be encouraged to post entries on the blog describing their learning. Their entries could serve as an insight to other students. As a result, a learning community is created.

Wikis

A wiki is a collaborative tool which helps groups to create and maintain work together. They are the ideal tool to implement the baseline of the outcome of elementary activities initiated by the teacher such as: exploration, researching information, experimentation and formulating hypothesis
Example of implementation: after reading the material presented in the guiding principles, for examples for OJTIs, students have to present work searching for information based on their environment.
Partial and final results are put in a common document. Individuals are able to cooperate on this document together and thus learn from each other.

Podcasts

A podcast is a downloadable audio file that a user can listen to whenever they want. Most radio stations offer the facility of freely downloading recordings from past radio shows. Also, with text-to-voice software many people are now translating their documents into voice recordings and listening to them through their MP3 players while taking a walk or doing another activity. In learning, podcasts could complement CBTs by allowing the student to download certain information which they could hear wherever they want to [the equivalent of printing information for future reference].

Webcasts

Webcasts are similar to podcasts, but include sound and video. The advantage of a webcast is when a presentation or a lecture is recorded and then sent to students to watch at their own convenience.

2.10.3 Conclusion: methods and tools

In the section above, a number of methods and tools were presented. The aim was for the e-learning developer in the ATM environment to take note of these tools and of how they could be used effectively in e-learning in our domain.

2.10.4 Some final instructional design tips

In the final section of this chapter, a number of extra design tips will be made. These tips are of a generic nature and should be considered in the design stages of e-learning content development.

A. Think interactive

The element of interactivity has been mentioned implicitly and explicitly in this chapter. It is important to remember that whenever possible the user should be given the chance to interact with the module. We believe it is better because students stay attentive and feel an active part of the training. However one word of caution is not to overdo it. Interactivity should serve the content and learning objectives, and not vice-versa. A balance needs to be sought so that the interactivity enhances the learning experience.

B. Consider maintenance issues during design

In the design of an e-learning module, those involved should already have its maintenance in mind. The domain might evolve, or usage and feedback could generate comments to improve the module.

Maintenance might be needed from a pedagogical as well as from a technical viewpoint. It is important to consider both aspects at the design stage so as to avoid extra work in the future.

C. Avoid meaningless learning

Here are some tips on the subject:

- keep text to the minimum so that the task of reading does not become too taxing[11];
- keep things simple, clear and precise;
- use visual and audio media so as to allow the users different 'processing' mechanisms to share the learning workload;
- given that different people learn in different ways it is impossible for one style to satisfy everybody, so vary the approach;
- only use media where necessary, to achieve the learning goal and not to provide glitz;
- avoid overusing visual effects for the same reason as above.

[11] Definition of taxing: burdensome, wearing: *a taxing business schedule*

D. Use a multimedia repository with your team

A multimedia repository stores and organises all the multimedia assets you and the rest of the development team can use to develop modules.

The advantages are:

- Contents for the entire department are centralised at a single point.
- It provides security regarding access to data so that the backup can be better controlled and personalised, as well as providing faster access.
- A single repository is able to be used as a test bank and therefore checking its solidity.
- It saves time in the development phase as the same work does not get repeated.
- A common library (graphics, images, buttons, scroll, menus, etc.) facilitates the job and enables a common style to be developed, both from the design point of view as with the programming.

The most significant disadvantage is that a repository is generally centralised, the availability of the data is thus dependent on the availability of the hardware. A number of contingencies should be found to avoid loss of data due to hardware failure.

E. Find the right balance in who controls the learning

Letting the learner control their learning is an interesting issue which can conflict with the trend that sees training managers wanting to control the learning path within a module.

There are occasions when it is helpful to devise an activity using an approach where one bit of knowledge is used to provide the platform on which greater understanding of the next will be based. In such cases, it would be detrimental to the process for students to jump ahead. However, the vast majority of people, from school days onwards, have studied from books in which they were free to look ahead and study as they wished. Thus some learners are perfectly capable of studying a course outside of the prescribed order, and regimented sequential courses hamper them and dim their enthusiasm.

Restricting access to topics sequentially actually deprives the student of control over their learning, and might be taken by some as a sign that 'teacher' does not trust or have much confidence in their desire to learn.

The advice here is to achieve the right balance between suggesting a structure while ceding control to the learner. They could opt to follow the structure proposed [in fact many do start a book from the beginning and go through it sequentially] while other more independent learners [or who already know part of the content] could jump straight into the parts that interest them.

2.11 Conclusion: Instructional Design

The aim of this chapter was to present the e-learning developer working in the ATM environment with a number of items that should be kept in mind when developing e-learning content. These items were based on good-practice experience in the domain.

The chapter started by focusing on the need to get to know the learner before starting to design the module. Together with the learner, an e-learning developer should also get to know what the target of the learning is. At this point a design process was suggested.

Within the design phase of an e-learning module, it was seen that navigational and look-and-feel issues are important elements to keep in mind. The password for this section was: "the_content_is_central", meaning that whatever a developer does with the interface, it should be to enhance the effectiveness of the content and not simply to make the look and feel stand out.

Following this, a number of methods and tools were described and other instructional design tips were given.

After having discussed instructional design, the next chapter aims at discussing the human aspects of the developer's roles and responsibilities.

3. HUMAN FACTORS

3.1 Overview

Educational software development is the marriage of two types of expertise - on the one hand educational expertise, such as pedagogy and knowledge of the subject matter, and on the other technical expertise, such as proficiency in the use of the software applications or the programming languages required to embed the knowledge into the software via a transparent interface in the form of readable and unambiguous text, challenging interactive learning tasks, and easily interpretable and well presented graphics.

There are no clear lines at the boundaries of each element - knowledge, pedagogy, interface, design, code, etc. - rather areas where merging takes place. For example, the programmer may have ideas for presentation which the SME had not thought of, or may understand that ideas for interaction may not work as the educationalist expected. On the other hand, developers are unlikely to understand the content as fully as the SME and, if left on their own, may design interactions that miss the point. To reflect this, the development team is best seen as a series of roles rather than as a group of people, since each individual is likely to be wholly or particularly involved in a number of different aspects of the development process. Perhaps more vital than the code, more vital than the knowledge, more vital than the skills of any of the participants, is the dynamics of the team. This is the point of the following chapter.

This section begins by looking specifically at the role and requirements of the developer. What makes a competent developer? Firstly, developers are advised to become aware of the limitations of their expertise, since doing so will show them how they can best integrate into the project. In short, unless they are actually from an educational background, developers should be aware that they are not pedagogues. A technically fabulous piece of functionality is worthless if it is based on an inaccurate model and does not reflect the knowledge to be digested.

Developers should, however, take on the control of the technical considerations of the project: the design, production and integration of the various media, the robustness of the deliverable (i.e. ensuring that it will run on the target platforms), and they should ensure that the design facilitates longevity - both in coding and in the maintenance of content.
Developers should also be aware that ATM has its own special needs based on its own peculiar knowledge base, skill set and language. Developing such awareness will allow them to help produce software that actually corresponds to the needs, work culture and expectations of their learners.

In order to maintain their effectiveness, developers should also keep up to date with technical developments. In some specialities the world remains generally stable and knowledge, once learned, can be good for many years. In the world of computing there is a rapid turnover of technology and it is important for developers to keep abreast of new software releases and have an overview of how the ‘game’ is changing.

With regard to the roles and responsibilities that are required with the team, it is important that all the players understand that their actions will always affect other members of the team, and act appropriately. For this reason, they should be aware that their own specific technical language may not be readily understood by other members of the team. They should take specific care when others are trying to explain matters to them and should not be afraid to probe for a fuller explanation of things they don't fully understand.

Given the fact that roles merge, it is likely that at times members of the team will be required to work towards the edge of their areas of competence and it is therefore helpful for people to try to understand not only their own role but also that of other members of the team. In this way, they will be less likely to step out of that realm altogether, thereby causing friction within the team, and will be more likely to understand why other members of the team might be behaving in a certain way.

The roles to be covered include not only the obvious ones of designer, developer, pedagogue and SME, but also quality controller manager and project manager, both of whom have their own specialist skills. The section closes with some advice on how to work in teams.

Finally, the relationship with the customer, which is touched on in many places, is addressed specifically. The customer may or may not actually have a clear idea of what they want. In reality, it is frequently the case that the customer will use early consultations with the development team as a sounding board on which to work out their ideas on this! Be also aware that at times the customer may have very close links with or even be integrated into the team, as an SME for example!

At the end of the project it will be the customer who decides if it has been a success. The developer should listen to the customer and try to provide what they request. However, the developer should also advise the customer as to what is realistic and feasible and when advising on different solutions should, if possible, demonstrate one!

The project plan should include dates when the customer can be informed of, or shown, the latest developments. These dates should be adhered to. If projects are delayed, partial products should be presented so that the customer is not left completely empty handed. If nothing else, this might also lead to an agreement to modify the project to take into account issues which were not at first fully understood or predicted.

When developing the software, all parties should keep the end user - their profile and their needs - in mind, and this should be reflected in a positive attitude on the part of the developer towards the end-user and the deliverable. Finally, anyone who wishes to comment on the product should use it first! Enjoy this chapter…

3.2 Introduction

In this chapter we will start with a word of warning; that we developers are not always pedagogic experts, and when we are not, we should take advice from those who are. Then we will talk about the skills that an e-learning developer should have, including considerations of our working environment; ATM. The chapter proceeds with a description of the various facets of the developer's job and gives a brief description of each. Finally the importance of working in a team and the relationship with the customer will also be discussed.

But first, developers are not always pedagogues…

3.3 Developers must be aware they are not [always] pedagogues

In general, we e-learning developers are the experts in technology and perhaps in design and we know the methods and possibilities which are useful in an e-learning module. However, this is not enough to provide the most useful pedagogical solution for a specific problem. In our practical work we meet two kinds of customers:

a) an SME without any pedagogical background. He/she is an expert in the subject concerned but is not working as a teacher or trained to be a trainer. In this case we should concentrate on the main focus of the delivered content and find out what is important in order to avoid making animations or films which go off on a tangent, since both developers' resources and students' time are consumed with e-learning. If we are not pedagogues ourselves, clearly the role of pedagogue is missing from the project. Ideally the person in charge of the project should go out and find one;

b) a teacher or trainer should be able to provide not only a specific content but also a concrete plan as to how this can be offered to a student in an e-learning module. In this case we should retain our consulting activity in those areas where technical aspects are to be considered. Here we have the most comfortable situation, where we only have to develop what our customer already has in mind. Perhaps we need some time and sensitiveness to find out what exactly our customer expects. But we may not overrule their ideas as they are the pedagogue.

3.4 Competence

Relationship work/competence

We are professionals. No question. But what does that mean? We are able to produce source-code. Is that enough to be competent in e-learning development? How about design, creation of 3D-objects, filming, photography, sound, etc. There are many fields of competence which have to be considered by an e-learning developer. The following are competencies to consider:

- being able to work in teams;
- respecting plans and the client;
- being knowledgeable about multimedia and how these could be applied in on-line distance learning;
- being competent in instructional design;
- being competent in the programming languages we use to produce our modules;
- keeping abreast of technological developments;
- producing clear documentation and comments on the code we write;
- ensuring that the development studio we use is still available in the medium term;
- being able to support the software with which we develop our modules at least in the medium term;

- producing standard source code so that others can maintain our work in the future;
- ensuring that the software we use runs on the end-users' terminals;
- showing interest and learning over time the general issues surrounding the ATM environment.

There may be other competencies that e-learning developers need to show in particular contexts. The above points, however, are general competencies that all e-learning developers working in the ATM environment should have.

What should an e-learning developer in ATM know? What are the key knowledge areas without which they cannot competently perform?

ATM is a world with its own priorities, language, rules and people. So working as an e-learning developer in ATM demands some special abilities which are pointing directly in that direction:

a) The language is not simply English, it is aviation-English which uses specific phraseology and demands a very precise wording to avoid mistakes. We have to get used to this kind of language, its abbreviations, units of measurement and unusual spellings. For example, a programmer who is asked to follow an aircraft's track might be doing this in km without even thinking about a possible mistake here, or why should anyone spell the word "three" other than using a "th" at the beginning or still, why is hundred and twenty wrong and one two zero correct?

Developers have to get used to this unique world before they able to process simple ideas[12]

b) Sometimes SMEs who works in ATM are so embedded in their word that they assume that everyone else understands what they consider to be 'common sense'. Sometimes an e-learning developer needs to consider timings, priorities and regulations which have been designed to grant safety and avoid misunderstandings. A good ATC e-learning module contains these unwritten rules but most of the time, trainers who asks for e-learning will forget to explain those rules. Developers might not have a clue about all these things so their software might – while still being designed excellently – be a nightmare for an ATCO. Again, there is a vital need for further education and training before a developer is able to fulfil tasks in the ATM world.

ATCOs may be seen as a group which behaves differently from other groups even in ATM. One needs to consider that these people have been selected for the job because of their ability [which is further strengthened during training] to make rapid and precise decisions.
The technology they are used to at their place of work is most reliable and they do not accept anything which obviously has mistakes and needs additional hooks and bolts. So keeping that in mind, we have to make sure our software is working (!) and simple to use. An ATCO will rarely agree to go through a complex process of learning how to use our program before starting to learn using e-learning.

[12] An introduction to the ATM e-learning module is available from EUROCONTROL. This module gives a first and general introduction to the ATM world. – for more information email info-elearning@eurocontrol.int

Keep it simple and make sure it works – otherwise you will experience what controllers do: they reject your software in the same way that they would reject a certain aircraft for their airspace. And this decision will be made rapidly, precisely and will not be discussed!

Keep up-to-date

Technology changes and these changes occur faster and faster. One might successfully argue that the task of keeping up-do-date is more than a full time job in itself! However, we should spend time gathering information about the developing world of programming and also have an eye to the many products which are provided for e-learning. New techniques grow like mushrooms and one life is not enough to take them all on board. The challenge is to find out what is relevant and what we may disregard. We currently use Flash (not always the most effective tool) to write c[13]-like code in action-script, while adapting easy-to-use multimedia elements within one single SDK. So we have to find out which technology or technique best suits our needs and will be available for a reasonable period of time. On the other hand, we must ensure that we are not drawn in by every new trend which appears exciting and glamorous but turns out to be a waste of time and resources.

Advice: keep up-to-date – in an intelligent and effective way.

3.5 Various facets of an e-learning developer's job

An e-learning developer's job involves various facets. This section will begin by listing a number of them and will then describe each one of those listed.

The various facets of the job of e-learning developer are:

- designer
- programmer/hacker
- quality manager/beta-tester
- project manager
- contact point for the customer

This is not a complete list of facets within e-learning development. These five facets show a typical team which is able to fulfil the task of e-learning design and development. It is important to keep in mind that we do not necessarily need as many as five people to start the development of a program. People may step into more than one role. At the lowest level you might be forced to step into every of these roles because you are the only developer in your organisation.

The design specialist is able to identify the feeling that a certain colour combination will produce in the user. An e-learning developer should also be at least sensitive to design issues which include placing the various elements and objects on a screen exactly where they have to be. The design-specialist will be responsible for HMI-design and the choice of fonts and frames. There is a big difference between a text that is simply written on top of the main surface of a program and one which appears inside a kind of monitor shown inside this surface.

[13] C is a programming language

There are people who have the instinct to find the right design for a program and there are those purely technically minded developers, who stick to the fact that the information is available (no matter at what size, colour or font). A good design is like a fast road: it supports the task to transform knowledge from the computer into the user's mind. Make sure your design-specialist is familiar with the necessary tools for the creation and manipulation of pictures, designing 3D-elements and other related software.

According to some people, a good programmer is the only person necessary to develop good e-learning software. Of course we need to programme/code our ideas and yes, we need those hackers who get up at three o'clock in the morning to note some cryptic characters which will do fantastic things once they are compiled. If we find someone with this gift, someone who demonstrates this attitude, we should let them do nothing else but this. Give them a computer; allow them to work at home, keep them in crisps and coffee – this kind of human being will not stop working until the thing works. The only thing we have to take care of is the kind of comments and documentation we will find inside the code and parallel to the source-lines on a document called "software-description" or "source-code-documentation". Truly fantastic programmers hate these like nothing else in their lives – of course, because it keeps them from coding. Perhaps we might consider involving another person to support them in following up with documentation. In any case we need to make sure that the generated source code will still be of any worth if our outstanding genius is on leave or has left our organisation for silicon-valley.

Developers may not love quality-managers, because they have to stick to certain defined procedures and rules which interfere with the creative process of a developer's work. However, if no one pays attention to certain basic rules concerning quality-control, we will have to spend an incalculable amount of time putting different modules together, fitting different designs and sizes into one program, fixing interfaces and so on. Keep up your rules of programming and make sure that the programmers act accordingly – even if they comment that this is wasting a lot of time and effectiveness. The more developers hate their quality-manager, the better he/she is. The same applies for a good beta-tester. This person has to be able to look at software very cleverly to find its errors and at the same time they must be able to act like the most stupid user we can imagine, to do things that a developer never expected to happen.

Someone has to take the role of the project manager, which means following the whole process with one eye on the plan. Rather than looking for good excuses if a customer asks for the reason for the latest delay, we should keep in touch with our customer and regularly tell them about the present status of the project. To do this, we need to know this ourselves, even if we are programming alone. It is hard to identify the amount of work we already have done compared to what remains. A good project manager will be able to give a percentage of development at different situations (milestones). We can train ourselves to do this as we develop software and we will find that it is possible and necessary for our project and especially for our customer.

Finally, all the people involved need to speak to the customer with one voice. Communicating outside the team needs to be coordinated: we must not talk about for-loops and recursive programming to our customer, who should not need to have a clue about programming when asking for a certain e-learning module. Here we need someone who is able to talk about an engine without using words like valve, piston, carburettor or spark. This is possible and we should spend time and effort in learning this kind of communication.

Many errors in programs have occurred because of specifications which had a different meaning to the customer on one side and the developer on the other side. If we find someone who is able to understand what the user needs (is asking for) and who thereafter is able to translate this need into words which a programmer/developer understands correctly, this person should be our point of contact with the customer. This person also has to explain to them what can be done and what cannot be done. Very often ATM experts hope to find a magic wand within e-learning and that this new tool can put almost everything they had to explain in a very difficult lecture into a self-explanatory programme with special effects a cinema director would be proud of. The point of contact needs to explain to the customer that this is not the case!

A person needs to be strong in at least one of these items to successfully integrate an e-learning development team.

Another requirement this person needs to have is to be able to work in a team.

Efficient and effective teamwork takes place when everybody has a shared view concerning the definition and placement of the above-mentioned facets, as well as of the other roles in the module development project team [read the project-management chapter].

If we accept these roles and also accept those who fulfil their tasks accordingly, we save a lot of time and discussions during a project. Teams will get used to the process of looking after each other in a certain way. Having gathered some (positive) experience a developer will no longer “hide” their code. They would rather share it with others to get help and bring the whole team one step foreword.

Spending time on team-building exercises is en effective way to improve the quality and speed of software-development within the group. A good and well accepted coding method should be implemented and closely followed to ease the task of understanding each other and provide programs which are not dependant on one special person.

Share knowledge rather than keeping it as a secret which might improve your position within the team! Make sure that teamwork is better than one-man efforts which no one will be able to follow up. Good ideas and methods must be shared in order to multiply their benefit.

3.6 Relationship with the client

The next section deals with the importance of developing and maintaining a good relationship with the client.

3.6.1 Accept every request, even when you do not agree

Beware of accepting customers' requests blindly [e.g. when asked to do things which go against your principles or which would damage your image] because there might be implications on how third parties see you.

At the end of every project it will be for the customer to decide whether you have succeeded in developing an e-learning module in accordance with their needs.

Regardless of whether we had the best possible idea, or produced state-of-the-art programming style or design – users will decide for themselves what kind of job they think we did.

So, if they want green stars on a pink background, while it would be necessary for us to give a hint concerning colours and their meaning, we need to respect our customer's choice. [This does not mean that we should not be clear that if changes are received after the specifications have been established, it may have consequences on time/budget constraints].

Sometimes we have an idea for an alternative way to explain something or to reach a certain goal which is obviously what our customer is aiming for. In these cases good practice shows that a client will be happy about us demonstrating a better solution. However, if this does not match the user's idea of the requested software, we should not spend hours trying to convince them of our idea.

Developing and maintaining a professional relationship with our customer is a good platform for successful e-learning development and will keep us in line with progress in the developing world of ATM-training. Remember: No matter how good or even excellent we are, if no one asks for our services, there will be no field to make use of our profession. If e-learning was a chess set, then we have to consider the customer as our king. (We mean the one on our side ☺).

3.6.2 Stick to dates

"Hello, I just wanted to ask about the present status of your software-development, since we expected it to be ready six weeks ago!"

Awful, isn't it? We as the developers are pretty well informed about the present status of the project (well at least we should be) but how about those people who planned to use this software at a specific date with a group of students which will definitely not be delayed?

We should – rather than defining the one and only definite date of completion – define a reasonable amount of steps during our development, where we ask the customer to have a close look to the progress and actual state of his software.
Fix some dates in your diary where you invite the user to have a look, regardless of whether or not you believe that there is something impressive to have a look at. We should use two items to follow the progress of a development:

a) Define milestones which are dependent on certain functionality which indicates the progress and gives everyone an idea about the path we have followed and the what remains to be done.
b) Define fixed dates (weekly, monthly – whatever is convenient) where you are ready to inform your customer about the progress of the development. If these dates meet a milestone: congratulations, this is what we want. If the customer has no need to make use of such a date – fine, we all save time.

3.6.3 Deadlines: if the project is delayed, it is more beneficial to produce a partial product at the agreed date than not producing anything before the final version is ready

Finally, if there is a defined date of completion, make sure you will be able to deliver software. If necessary, hand out a program where some items are still not working – at least the customer has something to work with.

We must avoid situations where our partners have nothing, not even an idea about what's going on and when they will be updated, or the reason for any delay.

The original plans can be ok

If the original project is appropriately developed and defined, it will lead to a favourable result. The starting point for the development of a successful project will be an original project which shows clarity of ideas, objectives to be reached, appropriate resources, coordination between developers and which take into account the final goal. In some projects, those involved keep changing their mind about certain ideas and specifications. Often, a series of changes end up in a 360° turn, and back to the original plans. The result is a waste of time and resources. Once specifications and ideas have been agreed in the project plan, everyone should stick to them and make an effort to avoid changes as much as possible.

Programme with the end user in mind

This item has already been discussed in the instructional design chapter, however it is worth repeating from an e-learning developer's view point.

When dealing with any e-learning development project, it is important to bear in mind the main element – the final user to whom the product is addressed – because the long term success of the product will depend on their degree of satisfaction.
Designing something for yourself is different than designing it for someone else.

It is convenient to analyse the user, know his/her profile, needs, limitations and ambitions and work with all these elements to guarantee the success of the product. The chance of success will thus be greater than if you take for granted the needs of the product user.

Do not gold-plate your projects

Gold-plating is when we as developers give more [much more] than what has been specified in the project.

Assuming that the planning and definition processes have been well followed, then the specifications have also been agreed and a formal or informal negotiation process has taken place to determine what is to be produced versus what is to be received.

Gold–plating is an inefficient use of our resources as there is a high risk that we keep adding to the module beyond what has been agreed. What normally happens at the end of this is that we get a thank you from our customers. End of story. But in the process of getting the thank you we could have lost other opportunities with other clients or have delayed other running projects.

What is to be developed needs to be specified in the definition phase of the project. What is outside the specifications can only be included via negotiation and a change-management process. We should not agree to extra work just like that, even if at times it might seem hard to say no!

Developer's attitudes

Finally, it is important to remember that keeping a positive attitude can be of benefit to us as developers, to our colleagues and to our projects.

While it is understood that everyone in the team has different characteristics and different ways of behaving, the attitude of the developer in charge of the task is fundamental for the good operation of the final product. A good attitude will translate into a good job, therefore the developer's confidence in his/her product is fundamental. In this way, we transmit such confidence to all users to whom the product is addressed. Enthusiasm, a positive attitude and confidence in the product's value should be the starting point of any product.

On the contrary, a negative or sceptical attitude when tackling a project will lead to an unconvincing and therefore mediocre product. How would we be able to convince others about our product if we are not convinced about it ourselves? What about the morale of our colleagues?

3.6.4 Conclusion: human factors

The human factors chapter is about the e-learning developer as a human working together with other humans to deliver successful e-learning material. Within this rather informal chapter, the competencies of an e-learning developer, the roles they need to take within a development team as well as their attitude to team work and towards the client have been discussed.

The next chapter in this guidelines document will be dedicated to project management and e-learning development. Often the e-learning developer only gets involved indirectly in project-management issues. However, we felt that it was important for the developer to at least be familiar with the mechanisms governing the management part of development.

4. PROJECT MANAGEMENT

4.1 Overview

This chapter is about how to manage the development of the courseware not the actual content. Experience has shown that if there is no understanding of project-management processes within the team, wrong decisions can innocently be made which turn out to have unforeseen adverse results for the project. Project management advises the adherence to clearly defined processes as a method of preventing such problems occurring.

The roles, and participants' expectations of each other, have to be discussed and agreed, typical pitfalls made clear to everyone, and responsibilities explained, defined and accepted. Although the recommended processes might initially appear to be a time-consuming hindrance, the opposite is true, and by keeping on top of them things should run smoothly. However, it is true that their complexity should be proportional to the size of the project, and small projects should not be burdened with overly complex project-management procedures.

The section begins by listing all the potential benefits of different stages of good project management and backs this up with the reasoning behind them.
After this, the roles and responsibilities of the various 'players' are listed, be they individuals or specific groups, such as a steering committee. Also explained is the importance of establishing common terminology to avoid ambiguity and misunderstandings between members of the project team. The relationships between the various roles and how they should interact are discussed. Be aware that in this section the discussion of roles relates to the project management and how people should work together, not in relation to the creation of the actual course contents.

Subsequently, the various steps through which a project will pass are mapped out - these are the phases of courseware development. The first phase is the definition phase, the objective of which is to provide clear and precise documentation relating to what the end goal should look like (i.e. the content and form of the deliverables) and the steps to be taken to ensure that it does. Formalising these matters allows all parties to understand what is required of them and what can be expected from the others. It also allows the project manager to keep things 'on track'.

The project plan that results will include details on a variety of matters, such as a breakdown of how and when the work will be carried out, how much it should cost, and an analysis of fall-back procedures in the event of possible problems such as illness to key workers, agreed terminology to prevent misunderstandings between people carrying out the different roles, or delays due to technical problems.

Be aware that developers and ATM SMEs may be familiar with acronyms and concepts that are likely to be completely alien to developers and SMEs from different backgrounds.

The project plan should also include the acceptance criteria for development strategy and the acceptable limits of the end product.

Once everything has been clearly spelled out and accepted, the actual development and validation phase can begin. In this phase, project management is concerned

with monitoring the development process to ensure that quality is ensured and the timetable adhered to, whilst at the same time keeping the risks under control. The tasks assigned during the project definition are now in the hands of those involved in the development, and the task of the project manager is to support and monitor those involved in that process.

The validation phase must have a clear set of criteria against which progress can be measured. Towards the end of the development process (but not exclusively) validation can be carried out by conducting usability tests on the software in which the testers voice their mental processing out loud while actually using the software. However, other criteria must also be assessed: does the software meet its objectives, is it sufficiently interactive, etc., etc?

Once all is well, project sign-off can take place. At this stage a formal evaluation should take place, and an end to the project formally signalled in some way. For the sake of improving the processes, all the lessons learned should be noted and fed back into the process documentation.

Once the courseware is released the maintenance process kicks in. This often represents a major and, initially, largely unforeseen commitment. The task is best approached by allocating certain responsibilities to members of the original development team - above all, the owner will have to play a committed role.

Finally, it remains to be pointed out that these processes are valid both for courseware produced internally and to that outsourced to commercial developers.

4.2 Preamble

Simple and/or short projects require short and simple project-management processes while complex projects might need you to use more of the tools described below. All the items presented in this chapter should be taken into consideration with this point of view in mind.

4.3 The benefits of having a project-management process in place

The benefits of having a project management process in place for defining, developing and maintaining e-learning content are listed below:

- A formalised project-management process will enable better control over the project thus making the process more time and cost efficient.
- Including quality and acceptance criteria in the definition and development processes adds to the probability of the product being designed and developed according to criteria which have been agreed beforehand by the client and the development team. This helps the project team to understand the quality standards and to deliver to these standards. It also enhances the probability that the client obtains a product which fulfils their needs.
- Because the process is being formalised, and because it requires a lessons-learned exercise at the end of development, it allows for continuous process improvement.
- The validation process allows the project team to keep in line with the users' requirements. This process is an extra check to keep the module as close to the end-user requirements as possible.

- The maintenance process aims to keep modules alive, up-to-date and usable for as long as possible. In this way, it extends the benefits of the initial investment.

Having a formalised process for the definition, development, validation and maintenance of e-learning modules will bring added value to the work done and to the end deliverable.

Roles and responsibilities while designing, developing or maintaining an e-learning module

The very first important item in any project is to know who else is involved in the project and what their role, responsibility and expectations from the project are.

An e-learning developer normally works within the project team, the typical members of which have been described in Chapter 1, in the 'The design process' section. The developer should be aware that outside this team there are other roles that will influence the running and outcome of the project as well as the maintenance of the end-product. The description of all the roles and responsibilities involved in the definition, development and maintenance of an e-learning module can be found in appendix 5.

Good practice recommendation: use agreed definitions within the project

One of the key elements within a project is communication, and communication cannot take place unless we speak the same language. Within any sort of project, normally people from different backgrounds are working together. One way of making communication effective is to have a set of agreed definitions for the project within the project environment. These definitions can range from definitions regarding the content, to definitions which describe the process or the roles and responsibilities.

What is important is that these definitions are understood and agreed within the project team. Some of the definitions need also to be understood by the project stakeholders. On a similar note, if a component library [repository] is created for the project, then the project team need to agree what format to use and what naming conventions, etc.

Agreeing on such items up-front helps the work go more smoothly during development.

4.4 Project management for defining and developing and maintaining e-learning modules

In the next few pages, the tools which will allow a project manager to manage the definition development and maintenance of an e-learning module will be described. The items might seem drier and less 'sexy' than the ones described in the previous chapter. However, for a successful project, both the management and the content aspects need to work well. As described in the section about benefits, above, project management is necessary to ensure that what is delivered is in line with what was planned and that everything takes place in the most efficient and effective manner possible.

The e-learning module life-cycle has been divided into three phases:

1. Definition
2. Development and validation
3. Maintenance

4.5 Definition phase

The first phase of the process deals with the definition of the module from a project-management point of view. It will describe the elements and actions needed to manage this phase.

At least three documents should be available at the end of this phase and before the development of the module begins.

These are:

1. The project brief
2. A description of roles and responsibilities
3. The project plan, including the acceptance criteria

Defining the project and the product properly before the development phase begins is important for:

- base-lining the plan and the criteria so that the progress during development can be compared to these plans and adjustments be made in a controlled way. It will also help to identify at an earlier stage areas where things are not going as planned and take appropriate action (issue management);

- formalising the agreement between the sponsor and project team on what is needed and what to expect from the end deliverable while taking into account the end users and the clients' needs.

- it helps all parties to understand each others' needs and expectations.

4.5.1 Project brief

The project brief aims to give a high-level view of what the project is about and who is involved in it. This document is important because it forms the basis on which the project plan will be developed. [And the project plan will be used to control the rest of the project.]

The project brief should include:

- identification of the commissioning group, i.e. who is mandating the project;
- identification of the project sponsor, i.e. who is funding the project and representing the interest of the commissioning group;

- identification of the customers, end users and any other known interested parties;
- the composition of the project steering committee i.e. the committee responsible for directing the project;
- the project definition, which should include the project's:
  - background,
  - objectives
  - deliverable(s)
  - scope
  - and any exclusions which need to be explicitly mentioned
  - constraints
  - interfaces
- an outline of the business case (this should only be an outline -no detail is expected at this stage). The business case should at least answer the following two questions:
  - How does the project support business strategy?
  - What is the need for the project?
- any known risks (this should launch a risk log which should be maintained and updated throughout the project);
- reference to any associated products or documents.

4.5.2 Roles and responsibilities

The second part of the documentation is the one which describes the roles and responsibilities of those who will be directly involved in the project and, later on, with the product.

For more details on roles and responsibilities see the above section carrying the same name.

4.5.3 The project plan

Following a high-level description of what needs to be done and why (project brief) and who will do what (roles and responsibilities), a project plan needs to be developed. This will give indications of what is needed in terms of resources, quality and time, and will feed the acceptance criteria (see below). It will be used to clarify what is to be expected at the end of the project and to control the development phase.

The items which should help the preparation of the project plan are presented below. This preparation is the responsibility of the project manager, but should be done in consultation with the rest of the project team and other main stakeholders.

Items in the project plan [Details to be found in appendix 6]

- Work breakdown structure
- Time plan

- Cost plan
- Risk log
- Acceptance criteria

Some likely risks to be encountered during the development of an e-learning module

- Misunderstandings between the SME and developer because they speak a different language
- Misunderstanding between the SME and scriptwriter because an adaptation of class room to technology-led training is a cause for conflict
- Technological risks
- e.g. adapting for a 2k by 2k interface on a 1024 X 768 definition
- Specifications keep changing [due to other specifications outside the project changing e.g. the development of a module describing a new human machine interface or due to stakeholders changing their minds]
- Not keeping the end-user in mind
- New features being added during development[gold-plating]
- Rushing through the design phase
- Perpetual editing of the script [team never happy with the result]
- Underestimation of the time needed to develop certain items
- …

4.5.4 Base-lining the project plan

The WBS with the task descriptions, time and cost plans together with the acceptance and the quality criteria should be agreed between the steering committee[14] and the project manager. Once this agreement is made, these are base-lined for future reference, monitoring and control.

4.5.5 Tolerance levels and project issues

The steering committee should assign the project manager with a set of tolerance levels regarding duration, cost and quality for either the whole project or the individual tasks within it. Should the project manger, during the development phase of the project, foresee slippage outside the pre-set tolerance, he/she should seek advice from the steering committee for a decision on the way forward.

[14] See Steering Committee definition in appendix 5

4.6 Development and validation phase

4.6.1 Development

Once a decision has been taken to proceed with the project, and resources committed, the project-management team must be focused on delivery within the tolerance laid down.

This means controlled production of the agreed product(s):

- to stated quality standards;
- within cost, effort and time agreed;
- ultimately to achieve defined benefits.

To achieve this success, the project must:

- focus management attention on the delivery of the agreed product(s);
- focus the resources used during development towards this end;
- keep the risks under control;
- keep the project brief under review.
- carefully monitor any movement away from the direction and product(s) agreed at the definition phase to avoid 'scope creep' and loss of focus.

During the development phase, the tasks which were identified during project definition will be assigned to the entities responsible according to the WBS.

4.6.2 Monitoring and control

As mentioned earlier in the document, the project manager should operate within a predefined set of tolerances. Should the project manager notice [potential] slippage from the agreed set of tolerances, they will need to report to the steering committee with a suggestion for further action.

If the steering committee decides to continue with the project but with the revised conditions then the approved plan should be updated. The project manager will then monitor the revised plan.

4.6.3 Validation

Validation means ensuring that what is being produced satisfies the learning need in a way that it has been decided during the design phase [see the Instructional design chapter].

This validation should start very early in the project, so that any deviations can be identified early enough and corrected. For the validation to take place, a process needs to be followed. This process should have the following questions answered:

➔ *Which kind of people will be involved?*

- developers
- SMEs
- Ergonomics specialists

- Multi–circle learners
  - Learners involved in the process in the final steps
  - Internal evaluators
  - External evaluators

➔ *Usability tests*

- Definition of testing process for part of the programme

➔ *When and how often will the validation take place?*

➔ *Which kind of reporting forms will be used?*

➔ *How will the results from validation be fed into the development process?*

What to validate?

➔ *The elements present in the acceptance criteria, and:*

➔ *A set of questions regarding the content of the module.*

These questions should be of the order:

- Is the content correct/exact/accurate?
- Does it keep the end-user in mind?
- Is it relevant?
- Is it well pitched to the end-user's starting knowledge?
- Is it realistic?
- Is it interactive?
- Are the learning objectives covered?
- Is the language clear?
- Are there spelling and/or grammatical mistakes?
- Is the module easy to use?

How to carry out a validation exercise:

A good way of carrying out the validation is by performing usability tests.

This can be done by inviting the validators to a session in which they can be observed while using the module. The validators should be asked to describe aloud what they are thinking while using the module [both in terms of ergonomics and content].

4.6.4 The end of development and project sign off

Once the project plan is followed, all the tasks are executed, and the e-learning module is ready to be delivered, the development project is ready to be concluded.

A clear end to the project is always more successful than the natural tendency to drift into use and subsequent modification of the product. It is recognition by all concerned that the original objectives have been met.

It also helps to achieve the business objectives by avoiding wasted time and by providing a useful opportunity to reflect on achievements and experience as well as providing an opportunity to ensure that all unachieved goals and objectives are identified, so that they can be addressed in the future. (These could be for example captured in a 'parking bay' document for future upgrades[15].

Lessons learned

At the start of the project a lessons-learned log should be created. A note should be added to this every time the project team spot something about the management or specialist processes and procedures that either made a significant contribution to the project's achievements or caused a problem.

At the end of the project (once the product is delivered), a report should be produced based on the reports, including any views with hindsight on the project's management.

This marks the end of the development phase. Once the product is delivered, it needs to be maintained. The next part describes the maintenance mechanisms.

After the module is delivered and starts being used, a maintenance process must be put in place so that the module's shelf-life is extended for as long as possible.

The next section of this chapter describes a maintenance process that could be introduced.

4.7 Module maintenance

As introduced above, the maintenance process's main aim is to extend the life of a module and to ensure that both the content and the interface are correct, functional and meet the learning objectives.

An e-learning developer would be asked to execute the maintenance. However, the maintenance process might not be a direct preoccupation. It is nevertheless important to know that a process can be put in place to organise maintenance. An e-learning developer could be in a position to influence this process inside their organisation.

4.7.1 The module owner

Firstly, it is suggested that the module is assigned to an owner who would have the responsibility of triggering any maintenance needed on the module.

This person should be selected on the following criteria:

- Preferably a user of the module (e.g. an instructor or a tutor who will use the module as part of a course he/she provides)
- Preferably have been involved in the definition and/or development of the module
- …

[15] For a list of items that the project manager needs to consider during project sign off, please see appendix 6

Note: The selection of a module owner should be done on a 'best fit' basis. If no one answering to any of the criteria above can be found, a person who at least has basic knowledge of the subject matter used in the module should be found.

4.7.2 Three different type of maintenance: corrective actions, updates and revisions

For an e-learning module, three types of maintenance actions are identified:

1. Corrective actions
2. Updates to the module, excluding content
3. Revisions to subject matter

While corrective actions will aim to cater for changes within the module after the identification of errors (e.g. spelling or grammatical mistakes, fundamental mistakes in the content, problems with the navigation within the module where the learning is severely impaired, etc.), updates and revisions deal more with the extension of the module's life-cycle and thus more fundamental maintenance of the content (revisions) and of the display (updates).

It is believed that most of the corrective actions required would take place in the first year(s) following a module's launch and after main updates and revisions. On the other hand, because a thorough preparation is needed beforehand and also because of resource and planning constraints, updates and revisions should be planned periodically over an extended period of time (e.g. every three years for revisions and every five years for updates).

Nevertheless, should the module owner at any time feel that the module needs an update or revision; he/she should present their proposal to the original project sponsor[16] for further consideration.

Corrective actions

Even after a rigorous quality check during development (validation), some mistakes could still have crept through into the final version. In this case, the mistakes should be corrected in a timely manner so that users retain their confidence in the product.

Three corrective actions have been identified:

1. Correction of bugs: the software does not function according to the specifications
2. Correction of spelling or grammatical mistakes
3. Fundamental mistakes in the content

The focal point for feedback gathering will be the module owner and this person should organise the feedback gathering. The corrective actions should preferably be performed by the entity that developed the module in the first place. When outsourcing, if possible this activity should from part of the after-delivery support service of the contract/agreement and should be agreed before the development starts.

[16] See appendix 5 for the definition of the project sponsor

Note for the module owner: during feedback gathering, a lot of ideas for future development and improvements in the module may pop up. It is suggested that a document where all these ideas are 'parked' is created. This 'parking bay' could be one of the references used in preparation for an eventual update or revision.

Updates to the module, excluding content

The module may look outdated even if its content is accurate. This may be due to:

- the look and appearance of the interface;
- the ageing of the standards on which the module was built;
- an original low cost (or time) development which produced a module only acceptable as a first version to be quickly upgraded.

Revisions to subject matter

There may be different reasons as to why a review of the module is needed.
These could include for example, a number of ideas to bring new content or to expand the scope of the module (expanding on the learning objectives), the subject matter becoming out of date, the inclusion of exercises which would enhance the module, etc.
At this point, a decision to revise or remove the module concerned needs to be taken and the final decision lies with the project sponsor.

As with the other maintenance actions, the module owner is the person responsible for monitoring changes in subject matter and flagging the changes to the appropriate entity.

4.7.3 Conclusion: maintenance

The development of an e-learning module needs considerable investment. To get the most out of this initial investment, it must, thereafter, be maintained.

Based on past experience, often, when a module is not assigned to a module owner, and when no maintenance mechanisms are in place, it is forgotten after a while and not used. This brings loss of the potential benefits which the module could still produce if it is maintained. The cost/benefit ratio of maintaining a module is quite attractive.

This maintenance mechanism aims at keeping the module alive and usable for as long as possible thus bringing added value to the e-learning portfolio.

4.8 Other considerations on project management

This last section of this chapter will include a number of further considerations on project management which have not been included in the sections above.

4.8.1 Project management when outsourcing all or part of the development

For outsourcing parts of an e-learning development or even the whole project one generally has to follow the rules stated within the process and the remarks on project management defined above.

In addition one has to keep in mind that most e-learning development companies do not have the same subject expertise and mindset as a development team which has been doing projects in ATM e-learning. When this is the case, the organisation should make an SME available to the company so as to reduce the risk of missing the point.

4.8.2 Communication when outsourcing

Communication between the outsourced team and the subject matter experts can be quite difficult owing to the different mentality which can be seen quite often within ATM experts and “the outside world”.

In order to allow both sides to fully understand what the other side is trying to communicate, a special verification process has to accompany all communication to ensure that both sides have really understood what the other side was trying to put across.

The project manager has the responsibility of making sure that this is done and acting as a translator and moderator between the two.

4.9 Conclusion: project management

Having a process for the definition, development and maintenance of e-learning modules and courses will bring added value to their life-cycle and to the investment of constructing these modules and courses.

Each phase of the process contributes differently to this added value:

The definition phase facilitates the understanding of what is expected from the module once it is finished and which process will be used to arrive at the final version.

The development phase brings better control over the execution of the project. It also assures that the product delivered is in line with pre-defined criteria.

The maintenance phase keeps the modules usable and up-to-date for as long as possible, with the aim of increasing the cost/benefit ratio.

The next chapter will provide a number of best-practice principles on technical aspects that an e-learning developer in ATM should keep in mind.

5. TECHNICAL ASPECTS

5.1 Overview

This chapter is aimed primarily at people who are starting out as developers and who have had little technical training in software development. This might be the typical situation of someone from an ATM background looking to change roles within their organisation. Nevertheless, the guidelines explained are programming fundamentals which developers of any level would be unwise to ignore.

Many software development applications provide great chunks of “black box” functionality via objects that can be incorporated into a program simply by selecting an option from a drop-down menu. This allows new developers to make quick progress with little knowledge, but they soon reach a wall beyond which it will be hard to progress without a deeper understanding.

Good programming is largely founded on three basic principles - knowing exactly what it is you are trying to do, getting there as efficiently as possible bearing in mind that every project on which you work should provide you with functionality (encapsulated in code or objects) that will be of use to you on later projects, and documenting/commenting code.

The first of these principles involves understanding basic concepts such as the use of algorithms to define code, and the use of layers, which are provided in many software development tools, and comments attached to code to provide structure and clarity to the development process. Other issues of note are the importance of the separation of media and the separation of content from interface.
The second principle involves a deeper understanding of the increasingly more sophisticated concepts of efficiency, modularity, maintenance, the use of code (or object) libraries and the design of intelligent objects that can more easily be controlled and reused.

Another important point to emphasise is the need to work to standards. Every piece of software is only part of a chain and nestles between the platform used to create it and the platform on which it will run. Since the sands of technology constantly shift, the idea is to maximise the life cycle of the software by ensuring certain kinds of compatibility - in short by adhering to standards.

The most relevant standard for e-learning delivery in the current environment is SCORM (Sharable Content Object Reference Model). The production of modules which are SCORM-compliant will allow them to be located on any learning management system (LMS) which itself is SCORM-compliant. This enables the software to be used on a variety of platforms. The ability to transfer the software to these different platforms is referred to as portability.

Related to that piece of advice is the need to work with development applications supported by the major players of the software industry. In doing so you will be able to upgrade software with the tools provided for you. It also makes sense to work with and master a few well chosen tools rather than exploring too many.

5.2 Introduction

This chapter is composed of a number of technical [programming] good-practice principles that an e-learning developer should keep in mind. An example of how a piece of code can be built based on these principles is also included in the appendix.

5.3 Principle 1: use layers

Layers are simply containers provided in certain programs into which an object or objects can be placed. Their use encourages discipline in the building of some types of objects, such as Flash movies or graphics, though not others, such as audio files or word-processing documents. They are a very useful development tool.

The purpose of layers is largely twofold:

1. to clarify and signpost the internal structure of the development file, so as to facilitate development and maintenance, and
2. to establish the visual hierarchy of objects on a page.

1. Layers can be renamed by the developers, allowing them to give them a label signifying their contents or purpose, so in effect, they can also be regarded of as a kind of commenting of the layout of the file, similar to the commenting of code, but they have more practical uses.
The grouping of certain objects into a layer, or even their specific and well reasoned separation into separate layers, can allow for objects that are not currently relevant to the task on which the developer is working to be hidden. Used effectively this can simplify the development task.

Development files, or even the objects within them, which are composed of only few components (other objects) may be easy to understand, but as they become more complex such transparency will quickly become lost to all but the developer himself (and perhaps with the passage of time even to them).

2. Layers, as can be gathered from their name, are automatically stacked by a program one on top of another. Programmers can alter the stacking order, either physically in the development file or sometimes dynamically at run time, so as to give precedence to a specified object(s) as required. Clearly objects on a higher layer may obscure objects on layers below them.

Note: HTML, layers and cascading style sheets.

Layers are a feature of some interpretations of HTML that cause confusion. Their purpose is to allow data to be in some way manipulated on an html page, for example hidden, shown or moved around. However, they are not a part of the W3C web standard and are not listed in the HTML 4.01 specification. More importantly, they are not supported by all browsers. The same effects can (should) be created in an HTML document using cascading style sheets (CSS).

5.4 Principle 2: portability

Question:

"Why was the gauge of the Spanish railway system chosen to be "six Castilian feet" -1674 mm?"

Answer:

"Because when the system was initiated, political instability in Europe meant that Spain felt vulnerable to invasion by France. As a result, in order to make a potential invasion more difficult, the Spanish government decided on a gauge different to the standard gauge already in use in France and most other western European countries (1435 mm). This is a decision that has, alas, caused problems for Spain in later generations. (Source: Wikipedia)"

Portability is the opposite of this; it is the idea of facilitating the transportation of something, for example a learning object, from one environment to another.

This document, being written in Word and saved in the Word Document format, is highly portable: it can be imported into some alternative word-processing applications, such as WordPerfect or some web browsers, such as Internet Explorer. The original can also be saved in a format that can be opened on a Mac. Documents written in more obscure file formats on more obscure software applications might only be capable of being viewed using that application.

Nevertheless, the real issue with regard to portability is LMS related. The LMS environment is somewhat unstable and the battle between providers to dominate a relatively new sector is still in flux. Any LMS chosen, from the hundreds available, must meet certain requirements (see SCORM) and courseware itself must be 'scormed' to allow it to be transported to a new LMS should the need arise, for example, if the provider of the LMS goes out of business.

5.5 Principle 3: commenting the code

Can't remember what that code you wrote six months ago actually does? Did you write any comments to help you? Well it could be worse, you could be trying to maintain someone else's uncommented code. Comments take up time, it's true - so this is going to be a trade-off, but especially if you work as part of a team, comments are very important.

More information and examples on what items could be commented on are found in appendix 7.

5.6 Principle 4: think about facilitating maintenance while designing and developing the product/make it easy to modify

The following six ideas items will show how we can facilitate the maintenance of the product:

1. Develop code libraries which can be shared throughout the site (perhaps even across different sites).

2. Look into object-orientated programming with its concepts of classes, hierarchies and inheritance.

3. Create intelligent objects (suggested activity - show two different controllable objects moving at different speeds and a slider to alter the speed(s) at which they are moving - then show the code/function the slider calls e.g. setSpeed (rate_or_value), then in the two different objects show different internal code to handle the command).

4. Never forget modularity - the old adage[17] holds good, if you find yourself rewriting the same few lines of code, it's time to rationalise things and turn them into a function. (see also point below) I.e. break larger steps down into smaller ones.

5. Rationalise the location of code - interpreted programs such as Flash allow code to be located almost anywhere, how can this be found by someone having to maintain someone else's code?

6. Build loadable components that can be externally loaded into different Flash movies, e.g. menu bars, caption bars, sound controls, interactive objects such as drag and drop objects, etc.

5.7 Principle 5: always know what a piece of code does, or else do not use it

The developer should know exactly what his code does before using it.

Sometimes, complex code downloaded from the internet can have unexpected results if we do not know exactly what it does!

5.8 Principle 6: separate media

It is better to create multiple smaller files rather than a large one.

For example, with a Flash movie download times become less critical since media (mainly audio and graphic files) can be downloaded and loaded into a movie in carefully designed sequences rather than 'all up front in one big lump'.
This helps to reduce the time that the user must wait while the file is being downloaded before they can interact with it.

The implications of this are, for example, that sounds should be loaded as .mp3 files and attached to sound objects rather than being imported into Flash itself and attached to movie clips. Similarly, large jpg files can be uploaded while the user is engaged in another task.

[17] Definition of adage: a saying that sets forth a general truth and that has gained credit through long use.

The other advantage is that different developers can work more efficiently on material to be delivered in the same main movie, i.e. one developer can be creating movie clips to be loaded into the movie created by another.

Others might agree on the final design for a graphic to be developed on the fly within a Flash movie containing text and images. While one specialises in the artwork the other can be preparing the text in XML.

5.9 Principle 7: concentrate on tools you know how to use

e.g. use mainstream software like flash

It is better to use one development software really professionally as an expert rather than using six different software-development kits without ever being able to use any of them properly. Different programs offer different strengths and possibilities – however, you are always good in what you use frequently. Even if the solution "might" be easier if you use a different method to generate it, stick to the software you know well to make sure there will not be any traps which drive you into unexpected trouble at a later stage.

By all means, the more programming languages you know well, the better! However it becomes more and more difficult retaining currency and fluency the more languages you know.

5.10 GOOD PRACTICES IN CODING

Introduction

Since there are thousands of books explaining how to write robust and efficient code and since the subject is worthy of an entire course, this section limits itself to outlining a few important points for people starting out to be aware of. Just grasping these points alone should help you to make quick progress as a developer. Furthermore, given that Flash has become such a popular tool, the comments that follow are very much related to it.

5.10.1 Use algorithms and start writing in pseudo-code

Write out the desired behaviour as an algorithm and convert it into 'pseudo code' before you write the actual code. Do this until you become familiar with the programming language you are learning, at which point you may skip the pseudo code step.

The reason for this is that it is important at the beginning to focus on what you actually want your code to do, and simultaneously wrestling with the correct syntax can be a damaging distraction. Therefore, for anything other than the smallest program always write out what you imagine the algorithms should be.

Return to these steps later if your program does not work as expected. By defining the actual algorithm you're more likely to write code which actually does what you think it does!

Finally, be aware that the process of writing the algorithm may cause you to question your initial assumptions and to rethink and tighten up your original design.

This is an extra bonus of this procedure - “did my first thoughts on this matter actually identify the best approach?”. Ask yourself the following sort of questions:

- *Does the design do what is required?*
- *Does the design do more than is required?*
- *Does the design actually do something that is slightly and subtly different from that which is required?*
- *Does the design do things which are similar to other code you have written?*
- *If so can that code be adapted to cope with this increase in scope?*
- *Does the design do more than one thing and if so might it be better to separate out the different tasks?*

Given that no piece of code will be entirely independent, you might also ask yourself if the current piece of design is actually dependent on another component that has itself not been well designed! And so on: the process is obviously an iterative one.

5.10.2 Efficiency

Experienced and extremely knowledgeable programmers can write extremely efficient code which taxes the processor as little as possible and runs robustly and quickly. The rest of us get by. In fact, this issue is not so important for interpreted languages, such as Flash, or small programs of the types which most Flash movies tend to be. Efficiency is extremely relevant to compiled languages or databases and large programs.

Why is this so? Because of modern monster processors. These can cope easily with less than optimally designed code, especially with interpreted languages where the code is compiled by the processor in smaller chunks as and when required rather than as a whole as the program is loaded.

Poor Flash code rarely seems to lead to noticeably slow running programs - although it may lead to programs that don’t run at all, or at best do something which you didn’t expect them to! As a result, discipline is still necessary and, if nothing else, it shows that the programmer has actually thought through an accurate algorithm. But don’t be afraid that your code might be less than perfect.

In actual fact, readability has now probably become more important than efficiency in coding, although it goes without saying that, as we intimated above, code that is rubbish will yield rubbish results (In technical terms, it will exhibit undefined behaviour).

5.10.3 Readability

Use meaningful variable names, use prefixes to signpost where code comes from (e.g. a library), avoid overly clever syntax but also avoid verbosity, comment code.

Why? Because someone may later be charged with the exasperating and unrewarding task of having to update or fix your code. Worse still, that person might be you. Unreadable code is the equivalent of an underground garage with no lighting.

5.10.4 Reusability

Reusable chunks of code, normally encapsulated as functions, prevent you from having to continually reinvent the wheel, and thus allow you to progress more quickly. This is particularly true, for example, with basic elements such as menus, buttons, sliders, etc. but also for all kinds of assets.

➔ See also modularity.

5.10.5 Modularity

Build larger programs by combining smaller steps into functions, this aids readability and maintenance. Avoid huge chunks of code.

Functions can be combined to produce reusable, more readable and efficient code. Of course, you can go one step further in Flash by building reusable components to do things such as control sound and video, but that is for another topic. In any case, good components are based on good modular code.

Having said this, it should not keep us from producing a source code of some thousand lines – as long as these are well sorted and organised in functions and single "blocks" which belong together. Avoid spreading the code all over the single movie clips in flash (whenever possible). It is better to have the code in one layer than to have one hundred single clips in the library which all keep a piece of code!

5.10.6 Libraries

If your site is likely to contain a number of Flash movies containing a certain amount of duplicated functionality, then it makes sense to extrapolate some of this functionality and store it in libraries. In this way you do not have to include the code in all the movies themselves; you simply link to the relevant libraries.

The advantage of this is that if a bug surfaces you can address it by repairing the code in the library and you will not have to go into each individual movie and update code there.

5.10.7 *Maintenance*

As you become familiar with the programming application, develop common locations where you keep code. You might also like to document these typical locations and try to keep their number to a minimum.

This is a task more easily carried out by experienced programmers, but if you really study the language you are using you should be able to rationalise this process. Knowing where to find code can make the task of maintenance a lot easier to execute. Remember also that this helps with the readability.

5.10.8 Try to create intelligent objects

If you can develop objects which know how to cope with their environment, you'll find that your programs become easier to manage. This is a feature of Object Oriented Programming. Having said this, the above philosophy does not have to be followed religiously in Flash to be of some benefit.

For example, a loadable menu bar that creates separate buttons for each menu item listed in an array. Instead of physically building the menu for each movie, just drop the object onto the stage and provide it with the data it needs via parameters to build itself duplicating movie clips at runtime. Compare this approach to physically creating them on the timeline. The diversity of behaviour of such objects is determined by passing parameters.

Example: coding

We thought that the principles above would be best described with a case example. This example can be found in appendix 7.

5.11 Conclusion: technical aspects

In this last chapter of the best-practices guidelines, a number of principles regarding technical aspects that a developer should follow were described. These principles, in fact, are very similar to those followed in programming, but the EDTF found that in everyday e-learning development work they are not always followed, to the detriment of efficiency and effectiveness.

The principles mentioned included coding, commenting on the code, using layers, thinking about the maintenance while developing and others. A number of examples have been attached in the appendices to illustrate these principles.

6. OVERALL CONCLUSION

The best-practice guidelines for e-learning developers that you have just read had the aim of highlighting the most important items that a number of e-learning developers in European ATM felt should be kept in mind to improve:

- efficiency;
- effectiveness;
- quality, and;
- cross-organisational e-learning development.

The document was specifically aimed at e-learning developers who are or will be working in the ATM environment. It was organised in three layers, each one more specific than the last. The first layer is the overview of each chapter which highlighted the main elements which should be kept in mind. The second layer was the detail found in each chapter. Generally the principles and good practices were developed in this part. The third layer contains further detail and examples which highlight the principles and good practices. This layer can be found in the appendices to this document.

The EDTF hopes that the reader found this document helpful. Should you have queries on the subject, please do not hesitate to contact:

development.elearning@eurocontrol.int

7. APPENDIX 1: DEFINITIONS

AICC
Aviation Industry CBT Committee. AICC is a standard for the passing of student progress data between e-learning content and learning management systems. Now included under the umbrella of the SCORM standard.

Application Service Provider (ASP)
An ASP (Application Service Provider) is a vendor supplying systems on an externally hosted basis, i.e. running from the vendor's hardware, not the client's. Authoring systems, learning management systems and virtual classrooms are examples of systems that can be hosted externally rather than 'inside the firewall' (on a client's own network).

Asynchronous communication
Communication where there is a delay between message and response. Examples of asynchronous communication include the post or, in e-learning terms, email, discussion forums/bulletin boards and blogs.

Authoring system
A software tool designed to facilitate the design, editing and assembly of e-learning course materials.

Bandwidth
Bandwidth is the data transmission capacity of a network, usually expressed in kilobits per second (Kbps) or megabits per second (Mbps). At one extreme, a low-end modem might have a bandwidth as low as 14.4 Kbps, whereas the Internet backbone (the network of major Internet connections) will operate at a minimum of 45 Mbps. A typical corporate network has a bandwidth of 10 or in some cases 100 Mbps.

Data-transfer speeds increase with bandwidth. With a sufficiently high bandwidth it becomes feasible to transmit high-quality audio and video on the Internet or an Intranet.

Beta
A beta version or release of computer software usually represents the first version that implements all required features although additional features may be added. It can be unstable (although this not necessarily so) and not yet be ready for release. Some developers refer to this stage as a preview or a technical preview. .
Blended solution: A solution that mixes a number of different learning media, such as face-to-face learning, online media, books, CDs or the telephone.

Blog
Short for 'web log'. A blog is a personal online journal which individuals use to record their thoughts and experiences. Blogs can be used within e-learning as a way for students to record and share their learning experiences. Blogs can be accessed on the World Wide Web or through special 'news reader' programs. It is common for readers to leave comments on blogs which the author will usually respond to.

Broadband
A high-bandwidth connection of at least 500 kilobits per second (Kbps), but typically now several megabits per second (Mbps).

Browse
A software application for viewing the World Wide Web. The most common web browser is Microsoft Internet Explorer, although others such as Firefox, Opera and Safari (Macintosh) are gaining ground.

Bulletin board
A shared messaging facility in which responses are posted to discussion topics or 'threads'. Bulletin boards can operate as a feature within a web site or with email messaging.

Chat room
A facility whereby messages can be exchanged between participants in real-time. Most chat is in the form of text, although the technology can be extended to support audio and video.

Course
An e-learning course is defined as a collection of technologically enabled items which are grouped together to fulfil a learning need. An e-learning course can be made up of interactive modules, synchronous virtual classroom learning, wikis, blogs, discussion boards, etc.

Discussion forum:
Another name for a bulletin board - a shared messaging facility in which responses are posted to discussion topics or 'threads'. Bulletin boards can operate as a feature within a web site or with email messaging.

E-learning
E-learning refers to anything delivered, enabled, or mediated by electronic technology for the explicit purpose of learning (ASTD).

Flash
Flash is a graphics software package from Macromedia [now Adobe], that can be used to add animation and interactivity to websites, without placing heavy demands on bandwidth. To view Flash in their browsers, users must have the correct version of the Flash plug-in installed on their PCs.

Formative Assessment:
A verbal or written factual assessment given for personal development purposes which should have an important and lasting influence on individual abilities or attitudes. Should be applied to the attention or use of one person in particular. (Source: EATM glossary of terms)

HTML
HTML stands for Hypertext Mark-up Language, a world-wide standard for describing and formatting web pages.

Instant messaging
A facility which allows Internet users to communicate together in real-time. Instant messaging programs, from Yahoo!, Microsoft/MSN, AOL, Google, Skype and others, let the user know when their contacts are online and available to be contacted. Instant messaging is typically textual, but can also use voice (called VOIP – see below) or video (using a webcam).

Instructional design
also known as instructional systems design is the analysis of learning needs and systematic development of instruction. Instructional designers often use instructional technology as a method for developing instruction. Instructional design models typically specify a method, that if followed will facilitate the transfer of knowledge, skills and attitude to the recipient or acquirer of the instruction. (Source: Wikipedia)

Instructional Technology
is the theory and practice of design, development, utilisation, management, and evaluation of processes and resources for learning. (Seels & Richey, 1994).

Java
Java is a programming language, created by Sun Microsystems. Small Java programs ('applets') can be downloaded from a web server to add functionality to a web page that would not be possible with HTML, such as the creation of a graph from user-supplied data.

JavaScript
Despite its name, JavaScript is not really anything to do with Java. Both are programming languages that can be used to augment HTML, but whereas Java programs are loaded as separate files, JavaScript is inserted directly into a page's HTML code.

LCMS
A learning content management system. An enterprise software solution for the creation of both formal and informal digital learning content and the delivery of that content on an individualised basis. Now often included within the functionality of an LMS (see below).
Learning Objects An entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning. Source: IEEE Learning technology Standards Committee: http://ieeeltsc.org

LMS
A learning management system. A system for managing access to learning materials, facilitating communication between learners and tutors, monitoring learner progress and reporting on usage. Can also be called a VLE (virtual learning environment).

M-learning
Mobile learning. E-learning using mobile devices such as palmtops and mobile phones.

E-learning Module
An e-learning module is defined as an individual stand-alone package of a limited duration. An e-learning module can be considered as a building block which used as stand-alone or together with other modules and other features will make up an e-learning course.

Open source
Open source software applications are created on a collaborative basis by programmers working on a non-commercial basis. The applications are typically free or inexpensive to purchase. The best example of an open source product is Linux, an alternative to Windows. In the e-learning field a good example is Moodle, the virtual learning environment that you are using now.

Performance support
Electronic performance support systems provide information and learning materials at the desktop for use on a just-in-time, just-enough basis.

Plug-ins
Plug-ins are special add-on programs that extend the functionality of the browser. The best example is probably Macromedia Flash, which adds animation capability. Plug-ins are also required to allow the output of legacy authoring tools, such as Authorware and Toolbook, to be displayed in browsers.

Podcasts
Podcasts are recordings of radio programmes, interviews and monologues in MP3 or similar audio formats for playback on portable music players such as iPods (although they can just as easily be played back from a PC). Using news reader programs and music software such as iTunes, users can have new podcasts from particular sources automatically downloaded and copied to their portable players.

RSS
Really Simple Syndication. RSS provides a means for Internet users to subscribe to news feeds, blogs and podcasts, so that new postings are automatically retrieved and downloaded for viewing/listening.

SCORM
Sharable Content Object Reference Model. An initiative of the US Department of Defence, Advanced Distributed Learning (ADL) programme, but now broadly agreed across the e-learning industry as an umbrella for the various technical standards needed to ensure interoperability between e-learning content and learning platforms (LMSs, LCMSs, VLEs).

Streaming media
Streaming is a technology for delivering audio and video in real-time on web sites. Rather than the user having to wait for large audio and video files to download before they can be played back, streaming allows playback to commence almost immediately. Streaming media solutions are provided by Microsoft, Real Systems and Apple.

Summative Assessment
An assessment which represents a summary of the learner's attitudes and abilities over a period of time. This summation should be given in both a verbal and written form and must be factual in content. It should not be given by those responsible for coaching a particular learner but should be an independent appraisal by suitably qualified personnel. (Source: EATM "Air Traffic Controller training at operational units").

Synchronous communication
Communication that occurs in real-time. Examples of synchronous communication include face-to-face discussion, the telephone or in e-learning terms, text chat, instant messaging and virtual classrooms.

Virtual classroom
Software that integrates many real-time e-learning delivery functions, such as text chat, audio conferencing, electronic whiteboards and application sharing. Virtual classrooms are provided by vendors such as Centra, Interwise, Webex and Microsoft.

VLE
A virtual learning environment, a platform for delivering online learning. Similar to a learning management system (LMS) but not intended for managing all types of training and primarily used by colleges and training providers. Examples include Blackboard and WebCT (which have now merged), and Moodle, a free, 'open source' system. A virtual learning environment (VLE) is likely to include facilities for synchronous and asynchronous communication between learners and tutors, as well as means to share materials, submit assignments and maintain records.

VOIP
Voice over Internet Protocol. The transmission of voice over the Internet in instant messaging programs and virtual classrooms. VOIP is increasingly being used as an alternative to the telephone, particularly for long-distance calls.

Web conference
Meeting or conference between participants in different locations conducted using the Internet. The software integrates many real-time meeting and e-learning delivery functions, such as text chat, audio conferencing, electronic whiteboards and application sharing. Web conferencing software is often used to create Virtual Classrooms and is provided by vendors such as Centra, Interwise, Webex and Microsoft.

Wiki
A wiki is a web site that can be edited and maintained by its users. The best known example of a wiki is the Wikipedia, an online encyclopaedia which is being constructed by thousands of volunteers.

8. APPENDIX 2: ACRONYMS AND ABBREVIATIONS

ATC	Air traffic control
ATCO	Air traffic controller
ATM	Air traffic management
CBT	Computer-based training
CSS	Cascading style sheet
EDTF	E-learning developers' task force [European ATM]
HTML	Hypertext mark-up language
LMS	Learning management system
SCORM	Sharable Content Object Reference Model
SDK	Software developer's kit
SME	Subject matter expert
WBS	Work break-down structure
W3C	World wide web consortium
XML	Extensible mark-up language

9. APPENDIX 3: REFERENCES

Answers.com: http://www.answers.com

Boyle, Patil, Towle, (2003): “Development of the NETg learner profile instrument”, whitepaper, NETg, Thomson learning: http://www.netg.com/pdf/NETg_WhitePaper_Profiling_the_Adult_Learner.pdf

Brookfield, Stephen D. (1986). Understanding and Facilitating Adult Learning, Jossey-Bass Publishers, San Francisco, 1986.

EUROCONTROL, EATM, (1998-2000): Integrated Task and Job Analysis of Air Traffic Controllers, Parts 1-3

Kator (2003) “Quick tips in design” in ATPM e-zine, August 2003: http://www.atpm.com/9.08/design.shtml

Lacefield (2005): Adult Education Tipsheets, website: http://roberta.tripod.com/adulted/indexadu.htm

Spool, Jared M., Tara Scanlon, Will Schroeder, Carolyn Snyder, Terri DeAngelo. (1997). Web Site Usability. North Andover, MA: User Interface Engineering. (f you are designing web pages and are concerned about readability, this book is a must-have.)

TAPTrainer: http://www.taptrainer.com

The American Heritage® Dictionary of the English Language, Fourth Edition Copyright © 2004, 2000 by Houghton Mifflin Company. Published by Houghton Mifflin Company.

Will-Harris Web, Type 101, a primer in Efuse website: http://www.efuse.com/Design/web_fonts_basics.html

10. APPENDIX 4: EXAMPLE OF AN E-LEARNING SCRIPT:

Topic Name: Introduction

<table>
<tr><td>Page 2: How does this web-based training work?</td><td>Last edited 17.1.7</td></tr>
<tr><td>
Who is this module for and what is its structure?
On this page you can find more information on this module.

Who is it aimed at?
It is aimed at qualified pilots and controllers.

What are the main objectives?
The main objectives of this modules are:
to highlight the risks of communication breakdown
to assist in the reflection of why these breakdowns may happen, and
to suggest ways of how to overcome these difficulties.

How can it be used?
This module has been developed to:
complement the All Clear Toolkit, in which you will find a series of movies
and other material aiming at improving air/ground communications, and
be used as part of refresher training for pilots and controllers.

Ideally it would used in conjunction with a peer discussion session
on air/ground communications. In this case the module should precede the discussion.

The module could also be viewed as stand-alone material.
</td><td>
The purpose/role of this page is:

To welcome users and to orientate them to the courseware

* *

The ‘point’ is:

The point is to show them how it works and other information about the module – e.g. Objectives, duration, etc.

* *

The page/program flow is:

Start idea (and backward link if any): this is essentially a ‘program flow’ item designed to develop and maintain continuity
Development of theme: stating the objectives and relating them to each other in greater detail than the point, minor points will be listed
Description of processes: for example ideas for interactivity, graphics, audio-visual material and relationships between them and the rationale behind them, includes the relationship with the pedagogical aspects.
End idea and forward link (if any): a recap and continuity device

</td></tr>
</table>

Are there any special needs?

If you are accessing this module via the internet you will need broadband, i.e. a connection speed of 512kbs or faster. This is because the module contains a number of videos clips, which are an essential part of it, which you will not be able to view adequately using a typical 'dial-up' modem internet connection speed of 4.7kbs.

You will also need headphones or speakers.

How long does it take?

The average duration of this module is of 20-30 minutes, however be aware that the expectation game could become addictive and that you might end up spending longer!

N.B. **You do not have to complete the module in one go.**

If you have accessed this course using a password, then when you return to it, you will be taken to the last page you studied. If you have accessed the module from a CD or via EUROCONTROL's 'Open Access' page, then each time you return to the module you will go to the Introduction.

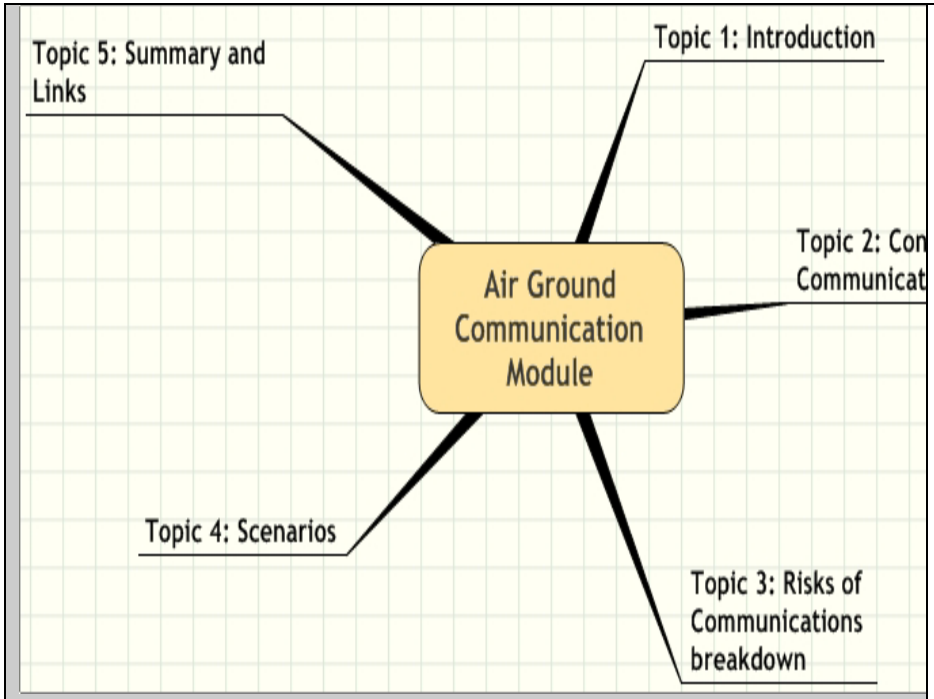
Problems with scroll bars?

All the contents of this module should be visible without you having to scroll.
If you wish to get rid of any scroll bars that may appear, do the following:

- ensure your screen resolution is set to 1024*768 px (optimum for this module) or higher,
- press the 'F11' key - this will hide most of your browser's tool and menu bars

Press 'F11' again when you want to restore your original browser settings.

The structure of this module The module is divided into three topics. The first topic is this simple two page introduction. The second topic, Communications, looks at the basics of good comms and asks you to reflect on some of the typical guidelines provided to help maintain high standards. The last topic, **Risk of Breakdown**, discusses some of the classic causes of communications breakdown and offers some suggestions as to how to avoid them. It includes video clips from the *All Clear Toolkit* and a game to help you analyse your own mind at work. Each topic is accessed by clicking on the captions in the menu bar above. We strongly recommend that you go through the topics and pages in order." When you have read the information select the next topic from the menu bar above.	
Since there will be more than 3 topics the last paragraph preceding will have to be extended as the final topics are developed. See below The fourth topic presents us a number of scenarios which will help us illustrate factors which can lead to communications breakdown. Many of these factors will be linked to the previous topic. The fifth and final topic will summarise the main items presented in this module and will give you links to additional resources. we hope you find this training interesting, fun, and a chance to reflect on how we can all improve our operational communications. [see voice over: AGC_T1_P3_MODULE_STRUCTURE_DESCRIPTION_CRYSTAL.mp3 it's a hyperlink]	



[We could improve on the display – maybe we could synchronise sound and animation?

Topic Name: Introduction

Page 3: What is the structure of this module?	Last edited 28.3.7
PAGE DELETED	The **purpose/role** of this page is: To welcome users and to orientate them to the courseware * The **'point'** is: The point is the idea that the student is meant to 'get'; it can be the objectives in expressed 'plain' English or the impact of the objectives, because very often the dry text of objectives hides the real issue which needs to be emphasised or unmasked. * **The page/program flow is:** • Start idea (and backward link if any): this is essentially a 'program flow' item designed to develop and maintain continuity • Development of theme: stating the objectives and relating them to each other in greater detail than the point, minor points will be listed • Description of processes: for example ideas for interactivity, graphics, audio-visual material and relationships between them and the rationale behind them, includes the relationship with the pedagogical aspects. • End idea and forward link (if any): a recap and continuity device
Comments moved to page above	

Topic Name: Communications – topic name to be confirmed later

Page 1: Air Ground communication is a tool	Last edited 3.1.7
Voice air/ground communications can be seen as one of the tools that is used by us c/p to fulfil our responsibilities. Like most of the other tools in any business, it can give enormous benefits if used well, but if misused, it could have dramatic effects. Let's consider the following analogy: Equations: Nail + shoe; nail + hammer; nail + hammer + closed eye Of course we don't expect to encounter such problems in our own work because we are so adept in using our communications tool. Let's remind ourselves of some of the fundamentals of good communications	The **purpose/role** of this page is: … * The **'point'** is: That air ground communications is a tool which needs top be used but which if misused could have dramatic effect. * The page/program flow is: Start idea (and backward link if any): Development of theme: Description of processes: End idea and forward link (if any):
better to say "adept at" Of course we don't expect to encounter such problems in our own work because unlike in the first equation we are given the appropriate tool for our communication purposes and as in equation two we are so adept at using it. Therefore, we ask again how can equation three ever happen?	

Topic Name: Communications

Page 2: Good communications game	Last edited 29.3.7
What makes good communications good? Having the right tool for the job is not enough. We need to use it with skill. Let us try the following exercise which will illustrate what we mean: [Left hand side underneath the text above, present the grid and the 3 task buttons as is currently. On the right hand side first present the following, then at the end of task 3 present the feedback] **Instructions** On the left you see a grid with numbers from 1 to 64. Underneath the grid there are 3 buttons for task 1, 2 & 3. Each task contains a set of instructions in audio format, so prepare your headset or speakers before clicking on task 1. You need to do the 3 tasks in sequence starting from task 1. **Aim of the exercise** Each task contains a starting number and a set of instructions that you need to follow in order to find out the destination number. Your goal is to find the destination number. Click on the number you think is the right destination. If you are correct, the blue background will turn to green, if not, it will turn to red! [leave the information above until the feedback is displayed] The feedback: We felt that task 1 was bet because it was: • clear • contained pauses so that we could catch up with instructions • logical • spoken at a good speed The other 2 tasks were less good.	The **purpose/role** of this page is: To give by example how good transmissions sound * The **'point'** is: To draw out some of the issues about techniques of good communications. * **The page/program flow is:** • Say that the tool is not enough • Give the user an exercise • Ask them to self evaluate • Give feedback – give the page's learning objectives • A small note to move to the next page.

Page divided into 2 columns

Left column:

1 map, 3 scenarios

The map would be that of a city [example: http://www.ipanema.com/citytour/images/mapleb.gif or something more cartoonish] create one of our own to avoid copyright?]. On the map there will be a number of dynamic points on which the students can click which we will use as destinations

The starting point is identical for the 3 scenarios

➔ **The goal is to identify the destination**

Scenario 1: A confused person giving a lengthy and non sequential description to arrive to a destination

Scenario 2: A person speaking fast but giving a sequential description to arrive to a destination with no pause

Scenario 3: A person speaking at an even pace giving a sequential description which is divided into pauses which the student can control.

Right column:

Reflection

Have a decision tree questionnaire:

Q1: *Which of the 3 scenarios, in your opinion, used the best communications techniques?*
Multiple choice: Scenario 1, Scenario 2, Scenario 3

<table>
<tr><td colspan="2">Page 3: Factors for good communication Last edited 3.1.7</td></tr>
<tr><td>Approved text goes here.

Make a list of factors that the student has to evaluate themselves on regarding good communications. The inspiration comes from HOTO.

The factors would be personal: starting with I

The factors will be positive [e.g. ‘When I feel under pressure I still manage to keep an even rate of speech’ and not When I feel under pressure I talk faster]

Put options: Never Sometimes Mostly Always

Get the factors from S. Garrets material, Helios Training objectives

Consider having some factors exclusively for pilots and others for controllers.

[maybe put a comment towards the end of the list that this could be used as a checklist for good communications…]

I use full call signs [pc]

I Insist clearances are read back in full [c]

Insist frequencies are read back in full [c]

Speak clearly – accent, pace, concisely,

I use standard phraseology where it exists [p/c]

I Listen carefully – write down requests and messages [p/c]</td>
<td>The purpose/role of this page is:

…

* *

The ‘point’ is:

….

* *

The page/program flow is:

• Start idea (and backward link if any):
• Development of theme:
• Description of processes:
• End idea and forward link (if any):</td></tr>
</table>

Cross-monitoring colleagues [p/c] ????Do not answer closed questions ????Ask only “open” questions Repeat clearances in full if in doubt [p/c] Transmitting at right time and in good time [p/c] I try not to interrupt colleagues when my message is not urgent if they are occupied with other work [p/c] If my colleagues put doubt on what I said or hears, I confirm with those concerned to take away the doubt [p/c Using headsets at least during busy and critical stages of flight (ideally for all clearances and below 10,000ft) [p] Use headsets all the time [c] At the end of the exercise, they hit the validate button. They will get the following message no matter what they have answered: The ideal answer is if you have marked ‘always’ on all the items. However the point is for you to assess the gap between your performance and the ideal and reflect on the importance it has for you. [You can ask your self do I have the opportunity to improve.] At the end of this introduce the next topic by saying that we will analyse some risks of breakdown in air ground communications	

Topic Summary

<table>
<tr><th colspan="2">Page 4: Topic Summary
Last edited 28.3..7</th></tr>
<tr><td>During this topic we have first considered that Air ground communications is a tool which enables us to execute our work.

Like any other tool, in order for it to serve us well, it needs to be used appropriately.

In the little exercise where we had to find the correct location as instructed to us. Here we investigated some of the ingredients that make up good operational communication:

• Speed of speech
• Consideration of the interlocutor[18] by
  o Grouping the instructions into digestible chunks and provide pauses between chunks
  o Being logical in the transmissions and building on the interlocutor's context
• Being concise – only communicate what is necessary for the task to be executed.

Finally on the 3rd page, we reflected on things we do [or do not do] in relation to air ground communications.

In the next topic we will explore the items that can cause us to have problems with operational communications from a human factors view-point.

When ready, please click on the topic tab: Risk of Breakdown, above, to move to the next topic.</td>
<td>The purpose/role of this page is:

…

* *

The 'point' is:

we summarise the topic and introduce the next

* *

The page/program flow is:

• Summary Page 1 topic 2
• Summary Page 2 topic 2
• Summary Page 3 topic 2
• Introduction topic 3</td></tr>
</table>

[18] Put definition of interlocutor in the glossary

Topic Name: Risk of breakdown

Page 1: The importance of communications [check spelling in actual page]	Last edited: 10.4.7
Communicating well is not an end in itself; it is a means to achieve safety and effectiveness. So, what effect does poor communications have on pilots' and controllers' workloads? [leave the rest as in actual page]	The **purpose/role** of this page is: … * The **'point'** is: to outline the typical risks **The page/program flow is:** • Start idea (and backward link if any): • Development of theme: • Description of processes: • End idea and forward link (if any):

*Define Bias: Inclination or prejudice in favour of a particular person, thing, or viewpoint. Source Concise Oxford English Dictionary

Put this definition also in glossary

11. APPENDIX 5: ROLES AND RESPONSIBILITIES IN AN E-LEARNING CONTENT DEFINITION, DEVELOPMENT OR MAINTENANCE PROJECT

Below is a list of definitions describing various roles and responsibilities within the project:

The commissioning group
is the group, team or other entity which commissions the project.
If the commissioning group is not directly the deliverable's client, it will be considered as the entity representing the client's interest.

The project sponsor
The unit or entity which is funding the project and which represents the interests of the commissioning group. This entity is the one which takes the paying decision of initiating the project.

The steering committee
The steering committee is responsible for directing the project (which is a management level above the project manager). The steering committee manages by exception. It monitors via reports and controls through a number of decision points. The project manager will inform the steering committee of any exceptional situation through issue reports.
The steering committee should be composed of a representative of the project sponsor and a representative of the entity which is providing the development resources.

The project manage
The person in charge of the tactical, day-to-day management of the project within limits set by the steering committee. It is the steering committee who designates the project manager.

The project manager will have to: [amongst other things]

- provide contact with the client;
- communicate with all the stakeholders of the project according to a communication plan;
- identify the needs;
- ensure that the agreed processes are followed;
- define the different phases of the project;
- ensure linear or parallel developments;
- coordinate with members of the team;
- monitor and control the execution of the project according to a project plan;
- define check points between phases;
- modify pace of development if needed.

The subject matter expert

The person(s) in charge of providing the subject matter for the development of the module.

The subject matter expert represents the source of knowledge. Different cases can be found:

➔ *The knowledge is well defined and has been pedagogically structured*

- Identification of general, terminal, intermediate and even elementary objectives.
- Interaction between objectives. Two cases can occur:

  a) This structure has been designed for learners with equivalent profile to the e-learning target.
  b) The structure was originally established for other types of learners. So an adaptation is requested (pre-requisite, mental processes, etc.)

- This situation will ensure the most efficient development.

➔ *The knowledge is well defined. Learning objects or media are available but without any real pedagogical structure.*

- The existing elements are to be taken into account to see the knowledge is actually transferred to learner, but the pedagogical strategy and the knowledge structure have still to be done.

➔ *The knowledge is "pedagogically raw". Sometimes, the SME is not a pedagogue, which means that the instructional designer will have to will have to adapt knowledge into a strategy.*

- The developer: the entity and person(s) in charge of developing the module in a computer language. This role is involved in design, selection of media and coding.

- The script-writer: The person in charge of writing the script. He/she needs to consult with both the SME and the developer and needs to take their needs and constraints into consideration. The script-writer also needs to be familiar with both other parties requirements in order to clarify possible communication problems between the SMEs and the developers[19]

- The e-learning module owner:
  The owner is the person in charge of triggering future maintenance of the e-learning module within the predefined mechanism.

- The end-user:
  The people who will use the e-learning module and learn from it.

[19] Note that one of the members of the project team needs to take the role of an instructional designer. Often this is shared between the script writer and the developer.

The project-management team consists of the project manager, the developer, the SME, the script-writer and the media specialist.

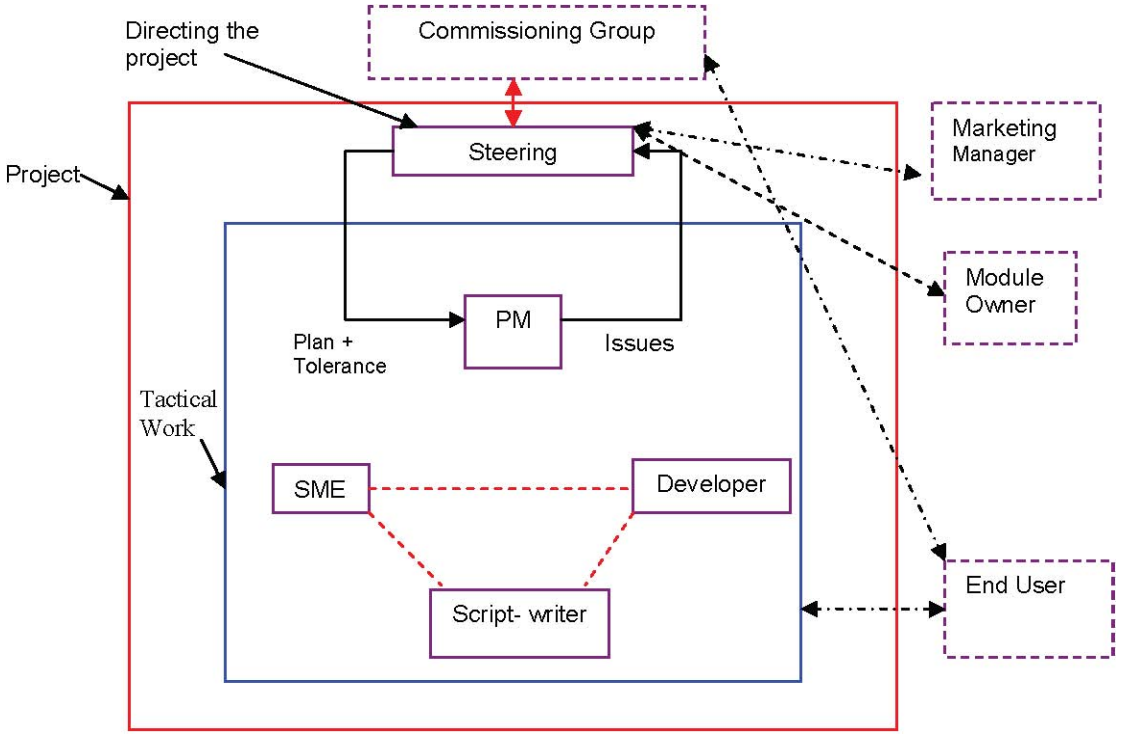
If some competences cannot be found in house, they must be outsourced. These competences should be in accordance with technologies and software available internally.

Note 1: One physical person may play several roles. Except:

a) *Members of the steering committee should not take another role from: pj team;*
b) *The project manager's role should not be doubled with that of the SME or the developer. This is because in case of conflict between the two, mediation would still be possible at project level.*

Note 2: If the commissioning group is not the direct client (for example the commissioning group identifies the need on behalf of a third party) some consideration needs to be taken to identify the client's needs and expectations.

The relationship between the different roles is illustrated in the diagram that follows:



The steering committee designates the project manager. Then, the steering committee with the advice of the project manager designates the rest of the project team.

When the roles are being established, a roles-and-responsibilities table should be produced. Below you find an example of this table: [this could go into more detail]

Roles	Unit/Section	Name	Comments
Commissioning Group			
Project Sponsor			
Project Manager			
Subject Matter Expert (SME) – Development			
Developer			
Training Designer			
Script-writer			
Owner			The owner accepts responsibility for future maintenance of the module. To be discussed. (May be co-ownership is better)

These roles and their responsibilities are also illustrated against different activities in the product's life-cycle in the next matrix:

Developing and maintaining E-learning modules **Activities**	**Commissioning Group**	**Sponsor**	**Steering Committee**	**Project Manager**	**S M E**	**Developer**	**Script-writer**	**Module Owner**
Mandating the project	**R**	**C**	**C**					
Defining the project	**C**	**C**	**C**	**R**	**I**		**C**	
Budgeting the project	**I**	**R**	**C**	**C**		**C**	**I**	
Managing the project			**C**	**R**				
Directing the project			**R**	**C**				
Maintaining	**R**				**C**	**C**	**I**	**T**

R = Responsible, C=Contributing, I = Informed, T=Triggering

12. APPENDIX 6: PROJECT-MANAGEMENT ELEMENTS

The project plan

Work breakdown structure

The work breakdown structure (WBS) helps to identify and organise all the different tasks which need to be carried out in order to deliver the product. These will include both specialist and management related tasks.

Breaking down the work which needs to be done into a distinct number of tasks is necessary for the building of a time and a cost plan. Every task should also be assigned to a responsible entity.

Some tasks which are normally necessary to produce an e-learning module are:

- Research
- Scriptwriting
- Implementation
- Validation
- Project management processes
- Testing
- Closing
- Lessons learned
- …

Time plan

A time plan should be built based on the tasks identified in the WBS.

Each task should be given a duration estimation and an analysis of the task dependencies should be made (e.g. if one task is dependent on another to start or finish, etc.)[20]. Current project-management practices should be used to construct this time plan.

A Gantt chart could be drawn to illustrate the dates at which each task would start and finish. Holistically, this time planning will allow to estimate the overall duration of the project and to plan for resources accordingly.

Once the time plan is finished and agreed by the steering committee and the project team, it should be base-lined as part of the project plan. The actual project progress should be monitored against this plan. This will allow closer control over the duration of the project and reactive measures could be taken by the project manager at an early stage to correct the path of the project, if a time slip is identified.

[20] The time plan in an e-learning module development project is more often than not non-linear.

Cost plan

The cost plan should also be built around the tasks identified in the WBS. Each task should be assigned a cost. Here as well, current project-management practices in establishing the budget needed to develop the e-learning module should be used. The advantages of having a cost plan are that:

1. The over all cost of the project can be determined;
2. Once the product is being developed, the initial plan helps to monitor and control the project's progress in terms of cost and allows for timely corrective actions if needed.

Risks and the risk log

The management of risks is one of the most important parts of the job done by the steering committee and the project manager. The project manager is responsible for ensuring that risks are identified, recorded and regularly reviewed. The steering committee has three responsibilities:

- notifying the project manager of any exposure of the project to external risk;
- making decisions on the project manager's recommended actions to risk;
- notifying corporate or programme management of any risks that affect the project's ability to meet corporate or programme objectives.

During the definition stage, a risk log should be created for the project.

The risk log should include:

- a description of the risk,
- a planned mitigation should the risk occur,
- an indication – if available, of when the risk is more likely to occur;
- a risk owner – the identification of the person who is best placed to monitor whether the risk is occurring and to take action based on the mitigation plan. The final decision of who 'owns' a certain risk lies with the steering committee. Steering committee members could be appointed as 'owners' of risks, particularly for those from sources external to the project.

Should the necessity of prioritising risks be identified, an impact and a probability value could be attributed to each risk.

Once a risk is identified and its possible impacts identified, a decision needs to be taken on whether to plan and resource for a possible occurrence of this risk.

Standard risk management practices should be used.

The risk log should be updated and monitored on a regular basis during the development phase.

Acceptance criteria

All the information which was gathered during the setting-up of the project plan will feed the acceptance criteria. This document aims at formalising the agreement between the parties involved, namely the project team and the sponsor, on what is acceptable in terms of development and end product. It should include:

- Target date for delivery
- Major functions
- Appearance
- Target users
- Performance levels
- Accuracy
- Availability
- Reliability
- Development costs
- Running costs
- Security
- Ease of use
- Timings

The acceptance-criteria document needs to be endorsed by the steering committee (the steering committee includes both client's and development representatives) before the actual start of development.

An example of a filled template of acceptance criteria is found in the table below:

		Comments
Target Date	25th June YYYY	Subject to confirmation by EL and SME
Major Functions	An e-learning module about level busts aimed at both pilots and ATCOs. It will be interactive and concise.	This needs to be elaborated further.
Appearance	In line with X Module	Subject to confirmation by EL and SME
Target Users	Operational staff highly knowledgeable with the matter which will be presented	
Performance Levels	The e-learning module has to respond quickly – lice ACAS module and has to interact with the user – visually and auditively	
Hardware	What kind of machine (processor, memory, graphic and sound performances, etc.) are supposed to be used by learners?) Hard disk capacity could be required	
Software	Eventually, which kind of software will be necessary to install on learner's computer? Limit new technology	
Network	What kind of connection? Will the sessions be running on internal network or will Internet facilities be used?	
Capacity	Not Applicable	
Accuracy	There shall be no mistakes in the content	
Availability	It shall be available to pilots and controllers	This needs to be discussed
Reliability	Reliability dependent on LMS server	To be confirmed
Development Cost	TDH	To be discussed
Running Cost	Same as above	
Security	Only available through login and password in Ims format – open to all in CD format	
Ease of Use	Standard e-learning module HMI	
Timings	See time plan	

Trigger to sign off

The project sign-off is triggered by the approaching end of the project or by it becoming apparent that the project is no longer viable for some reason.

The following is a list of items that the project manager needs to take care of during project sign-off:

- Ensure that the objectives set out during the project definition have been met.
- Request formal acceptance of the products from the steering committee.
- Ensure, where appropriate, that the arrangements for the support of the project products are in place (e.g. e-learning module on LMS).
- Identify any recommendations for follow-on actions.
- If the project has been closed prematurely, document what has been achieved and recommend a way forward.

Once the applicable items from the list above, and any other action(s) which is/are deemed necessary are performed, the steering committee needs to ensure that the products have been accepted by the project sponsor (including a cross check against the Acceptance Criteria list) who, as described earlier in the document, represents the commissioning group and the clients.

13. APPENDIX 7: CODING EXAMPLES

Example*: bubblesort*

The bubblesort is a simple (though inefficient) method of ordering a list, in which the highest or lowest item in the list continually 'floats' to the top (or 'sinks' to the bottom). Forget the fact that there are more efficient (though more complex) procedures for doing this. Before you try to write the code you must ask yourself "What am I trying to do"? This is not always obvious, and you may need some time to accurately formulate the algorithm.

Step 1: Analysis

Assume we want to write code to sort a list into ascending order, then:

- for example we want to convert the list [1,8,3,5,2] into [1,2,3,5,8].
- for each pass of the routine we want to shift the highest number in the list to the end
- during each pass of the routine we must successively compare one element of the array with the next and if the value is greater than the next value we switch them, then we move to the next comparison. I.e. compare item 1 with item 2 (if greater switch), then compare item 2 with item 3, etc.
- there will always be 'n' items in the list, and we will always have to make 'n-1' comparisons, but for each time we pass though the list (after the first pass) we can forget the last correctly ordered and highest value(s), thereby truncating the remainder of the list to be sorted.
- We need to repeat the procedure as long as there is more than one item in the increasingly truncated list to be sorted, which will mean we will need to perform the procedure 'n-1' times. You can check this by writing down the new order of the list after each pass and excluding the sorted value (shown here in bold). e.g. pass one yields [1,3,5,2,8] - pass 2 yields [1,3,2,5,8] - pass 3 yields [1,2,3,5,8] - although the list is now correctly sorted we wouldn't know this so we must perform one final pass of the list, - pass 4 yields [1,2,3,5,8]. So a five-item list requires 4 (5 - 1) passes.

From this we get the following algorithm:

Step 2: Algorithm

For each successive item in the list:

check to see if it is greater than the next value; i.e. 1>2; 2>3; 3>4; etc
YES then swap values: e.g. list = [2,1,3,1] is 2>1? YES -SWAP; now is 2> 3? NO; now is 3>1?
YES-SWAP. (i.e. let the highest bubble rise to the end of the list)

For each subsequent pass 'shrink' the amount of the list to be processed by 1 item (i.e. ignore the correctly sorted part of the list gathered form the most recent pass).

Repeat the process until there is no more of the list left to be sorted.

Step 3: The pseudo code (each programmer should develop their own)

Written in pseudo code this algorithm might look like this:

```
START
// initialise control variables
passes made = 0
max passes = list length - 1
max comparisons = list length - 1

process the list (max passes) times
        for each pass (for max comparisons)
                        // initialise control variables
                comparisons made = 0
                reference = 1



compare item (reference) with item (reference +1)
if item (reference) > item (reference + 1) YES:                 swap values
reference + 1
comparisons made + 1
comparisons made = max comparisons? YES:                        end of pass
                                                                else
                                                                go to compare

end of pass? YES: increment passes made by 1
                passes made = max passes? YES    return sortedList
                                                 else
                                                 max comparisons – 1
                                                 go to pass
END
```

Step 4: The actual code

This gives us the following ActionScript code for Flash. (N.B. The code could, of course, be written in many different ways, for example using different types of control loops such as ‘while’ statements).

```
function bubblesort(myList:Array) :Array {
        var maxPasses:Number = myList.length - 1;
        var comparisons:Number = myList.length - 1;
        var temp:Number;

        for (var i:Number = 0; i < maxPasses; i++){
                for (var j:Number = 0; j < comparisons; j++){
                        if (myList[j] > myList[j+1]){
                                temp = myList[j];
```

```
                                myList[j] = myList[j+1];
                                myList[j+1] = temp;
                            }
                        }
                        comparisons --;
                    }

                    return myList;
}
```

We can see that this code contains a couple of loops which provide concise code. There are however no comments so the readability is low. The variable names are at least vaguely intuitive and help to compensate a little for the lack of comments. The code would be an ideal function to be incorporated in a library, in which case it might be helpful to add a clue to this in the name. For example, if the code is to be stored in the library 'sortingFunctions.as' you might rename it sf_bubblesort() to provide other programmers with a clue as to where to find it. Now you can just link to this library when required. If a bug surfaces, fixing the code in the library will fix it everywhere.
[end of bubblesort example]

Commenting the code

Some things you might like to comment:

1. **Lines of code that affect other code not visible in the same code window or attached to different objects.**

   For example: A comment placed on the _root (or somewhere obvious) might read:

   ```
   /*
   To use the 'swap sentence' object included in the library of this movie, you must define a number of variables in an onClipEvent(load) handler for each instance dropped onto the stage. You will find the code in a readme layer within the object. Place the object on the stage, copy the code from the readme layer, select the object and paste the code. Uncomment it and set the variables as directed.
   */
   ```

   The code within the movie clip in the readme layer might read:

   ```
   /*
   // to use this object a minimum of three sentences are required.
   // the user swaps sentences in a table of TWO columns, or more,
   // by dropping one onto another. This makes them change places.
   // The user must read and analyse the table to decide where each sentence makes sense.

   onClipEvent (load) {
         var myCaption:String = "put the contents of the sentence here or refer to a variable elsewhere";
   ```

```
 // once 'done' is set to true, the object will become immovable
var active = false;
var done = false;
var i = 0;

// the object will automatically set its 'home' position when loaded
var start_x = _x;
var start_y = _y;

            // it is the programmers responsibility to initialise current_p and
             // correct_p appropriately
             // as objects are swapped they will swap their current_p values.
             // use check answers() to assess if each object's current_p equals
            // its correct_p.
            // N.B these values must be unique
            // 3 sentences? then the current_p values must be 1, 2, and 3, and
            // take care to start them out of position so that the object with
            // correct_p 1 starts in either current_p = 2 or 3.
            // Note that the correct_p locations can be anywhere in the table
            // and that the values represent an id and not a position.
 var current_p = 3;
 var correct_p = 1;
}

onClipEvent(enterFrame){
        if (active) {
                    // this will repeat until object is located correctly.
                    // controls inertial movement so that object decelerates
                    // as approaches final target position
                   // active will then be set to false
                    i++;
                      _root.ss_locateMe(this);
             } else if(_root.ss_dragging){
                    // highlights a valid target for the user to swap this sentence
with _root.ss_signalCanDrop(this);
            }
}

// you must also include the function checkAnswer() (found on the root) to
assess the user input.  Adapt the function as required.
*/
```

Note that not everything required to use this object is explained above, we are simply trying to illustrate the idea.

2. Links between objects whose behaviour is related

```
function g_displayWithAlphaFader(clipRef:MovieClip) :Void {
// the attached clip gradually increments the _alpha value of its
// parent and then deletes itself
 if(clipRef._visible == false){
                clipRef._alpha = 0;
                clipRef.attachMovie("fader","fader_mc",1);
                clipRef._visible = true;
               }
        }
```

3. Instructions for use for what is not immediately obvious

e.g. the code below is take from a function which controls inertial-based movement of objects on the stage.…

```
            // vary _root._MOVEMENT_INERTIA to alter speed of movement
            // suggest between range 0.2 to 0.5 if framerate is 24
ref._x += (ref.final_x-ref._x) * _root._MOVEMENT_INERTIA;
…
```

[end of commenting the code]

ISBN nr: 978-2-87497-001-6