



**Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom**

TELEPHONE:
INTERNATIONAL:
FAX:
E-MAIL:

**Cambridge (01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk**

Training

ANSAwise Training - analysis of feedback from courses

Chris Mayers

Abstract

Customers for ANSAwise training courses have expectations of the course content and presentation. These expectations have not always been met. The APM Training Plan assumes that some improvement to existing courses will be necessary and possible; it also proposes new training course development, both for customized courses, and also for public courses.

Improvements can be made most effectively with some changes to course content, course presentation, and course development procedures.

This document analyses the feedback from the ANSAwise training courses, and proposes a set of actions for improvement consistent with the overall APM Training Plan.

APM.1449.00.01

Draft

6th July 1995

Project Management

Distribution:

Supersedes:

Superseded by:

ANSAwise Training - analysis of feedback from courses



ANSAwise Training - analysis of feedback from courses

Chris Mayers

APM.1449.00.01

6th July 1995

The material in this Report has been developed as part of the ANSA Architecture for Open Distributed Systems. ANSA is a collaborative initiative, managed by Architecture Projects Management Limited on behalf of the companies sponsoring the ANSA Workprogramme.

The ANSA initiative is open to all companies and organisations. Further information on the ANSA Workprogramme, the material in this report, and on other reports can be obtained from the address below.

The authors acknowledge the help and assistance of their colleagues, in sponsoring companies and the ANSA team in Cambridge in the preparation of this report.

Architecture Projects Management Limited

Poseidon House
Castle Park
CAMBRIDGE
CB3 0RD
United Kingdom

TELEPHONE UK
INTERNATIONAL
FAX
E-MAIL

(01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk

Copyright © 1995 Architecture Projects Management Limited
The copyright is held on behalf of the sponsors for the time being of the ANSA Workprogramme.

Architecture Projects Management Limited takes no responsibility for the consequences of errors or omissions in this Report, nor for any damages resulting from the application of the ideas expressed herein.

Contents

1	1	Introduction
1	1.1	Audience
1	1.2	Scope
1	1.3	Status
2	2	Overview
2	2.1	Business aims and objectives
2	2.1.1	Technical aims and objectives
2	2.1.2	Proposed solution
3	3	Analysis of course feedback
3	3.1	Sources of feedback
3	3.2	Comparative material
3	3.3	Analysis of feedback
3	3.4	General feedback
3	3.4.1	Course structure
4	3.4.2	Course approach and focus
5	3.4.3	Course quality
6	4	Course content improvements
6	4.1	Understanding Distributed System Architecture - Day 1
6	4.1.1	Welcome (APM.1339)
6	4.1.2	Introduction to Distributed Systems (APM.1328)
7	4.1.3	Distributed and Networked Operating Systems (APM.1317)
7	4.1.4	Characteristics of Open Distributed Systems (APM.1318)
8	4.1.5	Templates for Distributed Applications (APM.1320)
8	4.1.6	Architecture for Open Distributed Systems (APM.1323)
9	4.1.7	Networking in Distributed Systems (APM.1331)
9	4.2	Understanding Distributed Systems Architecture - Day 2
9	4.2.1	Review of Day 1
9	4.2.2	Introduction to CORBA and DCE (APM.1322)
10	4.2.3	The ODP Reference Model (APM.1336)
10	4.2.4	Management of Distributed Networks (APM.1324)
11	4.2.5	Distributed Communications Techniques (APM.1340)
11	4.2.6	Distributed Database and Distributed Systems (APM.1332)
12	4.2.7	The Computational Model (APM.1327)
12	4.3	Understanding Distributed Systems Architecture - Day 3
12	4.3.1	Review of Day 2
12	4.3.2	Trading and Federation (APM.1330)
13	4.3.3	Service Quality in Distributed Systems (APM.1321)
14	4.3.4	Distributed Workflow Applications (APM.1319)
14	4.3.5	Security in Distributed Systems (APM.1334)
14	4.3.6	Services in the Electronic Marketplace (APM.1326)

15	4.3.7	Course Roundup (APM.1335)
15	4.4	“Building Applications with Distributed Objects” - Day 1
15	4.4.1	Welcome
15	4.4.2	Objects in Distributed Systems (APM.1350)
16	4.4.3	Specifying Services for Distributed Systems (APM.1348)
16	4.4.4	Remote Procedure Call in Distributed Systems (APM.1344)
16	4.4.5	The Engineering Model (APM.1351)
17	4.4.6	DCE Distributed Services (APM.1373)
17	4.4.7	The CORBA Object Management Architecture (APM.1345)
17	4.5	“Building Applications with Distributed Objects” - Day 2
17	4.5.1	Concurrency in Distributed Systems (APM.1365)
18	4.5.2	Building Applications with ANSA (APM.1366)
18	4.5.3	Naming in Distributed Systems (APM.1343)
18	4.5.4	Dependability in Open Distributed Systems (APM.1341)
18	4.5.5	Designing Applications with CORBA (APM.1352)
19	4.5.6	CORBA Object Services (APM.1349)
19	4.6	“Building Applications with Distributed Objects” - Day 3
19	4.6.1	CORBA in the Real World (APM.1379)
19	4.6.2	Replication Techniques for Distributed Systems (APM.1358)
20	4.6.3	Exploiting High Performance Networks (APM.1357)
20	4.6.4	Real-time Distributed Systems (APM.1353)
20	4.6.5	Multi-media in Distributed Systems (APM.1354)
21	4.6.6	Course Round-up (APM.1356)
22	5	Course presentation improvement
22	5.1	Session length
22	5.2	Discussion sessions
22	5.3	Slides
22	5.3.1	Slides quality
22	5.3.2	Time and content per slide
23	5.4	Course handouts
24	5.5	Course assessment questionnaire
24	5.6	Presenter preparation
25	6	Course development process improvements
25	6.1	Aims, objectives, and learning outcomes (AOLO)
25	6.1.1	Aims
25	6.1.2	Objectives
26	6.1.3	Learning outcomes
26	6.1.4	Proposal
26	6.2	Outlining and story-boarding
26	6.2.1	Strength of argument
27	6.2.2	Outlining
28	6.2.3	Story-board
28	6.2.4	Proposal
29	6.3	Course development teams
30	7	Course structure
30	7.1	Courses
30	7.2	Course modules

1 Introduction

1.1 Audience

APM Management

1.2 Scope

This document covers reviews of all chargeable training courses (including executive briefings, technical seminars, and so on) presented by APM. Currently, these are covered by APM project codes 70 and 71.

It does not cover other non-chargeable presentations (sales presentations, technical seminars, conference presentations).

This document covers the content and presentation of these training courses; course marketing is outside the scope of this document.

1.3 Status

This is a draft plan for approval by APM Management.

2 Overview

2.1 Business aims and objectives

This review of APM training activities has identified the following business aims:

- Train a second course presenter to give each course
- Reduce the effort costs in developing and revising course material
- Allow other organizations to present the course material (required for CNET - see [CNET Proposal])
- Allow other organizations to translate the course material into other languages ([CNET Proposal])
- Continue to improve quality of course delivery in line with the 'good/better/best' strategy given in [Training Plan], taking into account the urgency and importance of any changes needed
- Offer a range of new courses with different durations

2.1.1 Technical aims and objectives

This review of APM training activities has identified the following technical aims:

- Increase the technical content of the courses
- Update the content to reflect evolution in standards and products (particularly CORBA)
- Replace lecture content with practical examples
- Complete the documentation of course development processes (including estimation and monthly planning to smooth out development)
- Use the same review process as for ANSA Phase 3 deliverables
- Increase the reusability of course material across courses (particularly for induction of new sponsors)

2.1.2 Proposed solution

The proposed solution is as follows:

- Selective updating of existing course material
- Transfer of some existing course material between courses
- Development of some new course material (as proposed in [CNET Proposal])

3 Analysis of course feedback

3.1 Sources of feedback

This document analyses feedback from:

- courses participants, as filled in on the Course Assessment Questionnaires handed out on each course. A statistical analysis of this feedback for each has also been prepared
- Yigal Hoffner's attendance on the course "Understanding Distributed Systems Architecture", summarized in [Course Review]. This is the main source of feedback considered here. [Course Review] also includes some feedback from a paper review of "Building Applications with Distributed Objects".
- informal feedback from Ashley McClenaghan and Ben Crawford's attendance on the "Building Applications with Distributed Objects" course
- review comments made on the course material when originally prepared, but which it is was infeasible to incorporate at the time
- an ANSA Technical Committee survey for proposed new modules
- self-assessment by the course presenter

3.2 Comparative material

The [DCE Internals] course material was used as a comparison, as a source of best practice.

3.3 Analysis of feedback

Feedback indicates that improvement is needed in:

- the course structure and content
- the course development process, particularly in setting objectives and reviewing
- the course presentation

Localized comments on inaccuracy are not included here; it is assumed that these will be corrected when the individual modules are updated.

3.4 General feedback

3.4.1 Course structure

Each existing course covers the entire field of distributed computing, in progressively greater depth from course to course. The material is covered

'breadth-wise'. Customers would prefer 'depth-wise' coverage focusing on specific topics.

Specific topics that were mentioned repeatedly were:

- Network/system management
- Performance engineering

However, the courses should not focus on specific types of applications; it should cover generic techniques for building systems.

Customers would also prefer shorter courses (say, reducing the total course length from 6 days to 3 or 4 days).

Some of the material in the course is only indirectly related to distributed systems; participants commented that this was "interesting but not directly relevant". This should be extracted into optional add-on modules.

3.4.2 Course approach and focus

Technical material can be approached in different ways:

Table 3.1: Course approaches

Why	Business/technical motivation
What	Concepts, standards, products
Who	Organizations, vendors
When	Maturity
Where	Applicability
How	Methods, development process, techniques

Most of the courses currently concentrates on *why*, *what*, and *who*. For the courses to be useful, participants want to understand *where* and *how* to apply these concepts to their own projects; there is not enough material of this kind in the courses at present. (This also implies that the course must be better linked to software development/software engineering practices in life-cycle and architecture and design techniques.)

Technical material can also be focused in different ways. It is proposed to change this as follows:

Table 3.2: Course focus

Currently	Planned
Problems	Solutions
Models	Examples
General	Specific
Abstract	Concrete
Detail	Summary
Theory	Practice
Research	Experience

Almost all the course participants so far have had a strong technical background; currently the technical level of the course material is rather too

low, except in the theoretical sessions, where it is far too high. (These theoretical sessions should be eliminated, anyway.)

Focusing on implementations that are known to be successful (positive case studies) would build credibility.

There was no specific demand for training on new topics being covered in ANSA Phase 3. The focus should be exclusively on near-term technologies. It may be appropriate to cover mid-term technologies in a brief final 'vision of the future'. Long-term research topics are inappropriate.

3.4.3 Course quality

No attempt has been made to calibrate the course feedback against the 'good/better'/'best' strategy (for example good=3, better=4, best=5). Course feedback is subjective, and is best analysed for trends and deviations, rather than against an absolute measure scale.

4 Course content improvements

This chapter proposes improvements to existing courses by updating or replacing weak modules, and adding new modules. For a more detailed summary of the purpose of each course module, refer to the FrameMaker cover sheet of its document.

[Course Review] proposes that “Understanding Distributed Systems Architecture” be reduced from 3 days to 2 days. It is proposed that this be done, but only once planned course presentations have been given.

4.1 Understanding Distributed System Architecture - Day 1

4.1.1 Welcome (APM.1339)

This session is not a module; it is a welcome session that contains a brief road-map of the course. [Course Review] notes that the road map does not provide any insight to why the topics were chosen nor to their relative importance and relationships between them. In its current form the road map exposes the lack of coherence; there are too many parallel tracks.

[Course Review] also notes a lack of explicit statement of objectives (what will the listeners know/be able to do at the end of the course).

[Course Review] suggests that the road map should be moved into the first main session §4.1.2 *Introduction to Distributed Systems (APM.1328)*. This would have the disadvantage that

I believe that the problem lies in the course content, not in the way the road map is presented. Once the content has been improved, the road map will fall out naturally. (It would nonetheless be useful to fit the road map onto a single page.)

It is proposed that the revised style of Welcome session as used in [Impact of Distributed Processing] be reviewed and considered.

4.1.2 Introduction to Distributed Systems (APM.1328)

This module is a basic introduction to distributed systems starting from a telecommunications business perspective, and moving rapidly to a technical perspective.

[Course Review] states “I could not detect a structure in the slides” and “This part was very weak and also raised comments from attendee”. This indicates a much more widespread problem since this session is largely copied from existing APM introductory presentation material. It would seem that the introductory ANSA/APM/Distributed Systems presentation material is inappropriate or inadequate for this target audience. If so, this indicates a problem for sales/marketing presentation material too.

[Course Review] suggests that everyone at APM should be confident with this session and be able to present it; this is a reasonable expectation.

[Course Review] asserts that “Architecture for reuse is inappropriate. The architecture is for dealing with complexity.” This is not compelling enough; how does controlling complexity save money? - there is a much more obvious link between reuse and saving money.

One possibility would be to drop any mention of architecture on this introduction, and focus entirely on the problem rather than the solution.

Course participants have also commented that a 1-hour introduction is too long.

It is proposed that the structure and content of this module be separately reviewed.

It is also proposed that this module be presented following an ANSA project meeting. (Similar high-level/introductory presentations have been handled like this recently.)

4.1.3 Distributed and Networked Operating Systems (APM.1317)

[Course Review] suggests that this module is a ‘history story’ and should be introduced as such. Unfortunately, course participants are unlikely to be interested in the history, however relevant it may be; they are concerned with solving today’s problems, not understanding how they arose.

[Course Review] suggests aligning discussion of technical evolution (advances in hardware, software, and networking) with discussion of organizational evolution (office automation, groupware). This is valid, but coverage of organizational evolution may be sensitive.

The material on distributed operating systems should be dropped, as it is an approach that never succeeded in the marketplace.

The material on networked operating systems (NetWare, NT Advanced Server and IBM LAN Server) needs to be updated for the latest versions of these products.

[Course Review] says that the material on micro kernels and multicomputers is unnecessary. I feel it is important to cover these topics, since participants are likely to have heard them mentioned, but may be unclear about their meaning and relevance.

It is proposed that this module be updated.

4.1.4 Characteristics of Open Distributed Systems (APM.1318)

This module is a collection of four brief exercises exploring the characteristics of openness, scalability, diversity, and concurrency; characteristics that are inherent to distributed systems. It attempts to convince the participants that the characteristics of distributed systems are fundamentally different from centralized systems, and introduces the principle of reversed assumptions.

The exercise on openness has usually been successful, and makes its point. The other exercises are much less successful.

[Course Review] claims that the list of characteristics is incomplete. [Abstract and Automate] lists 10 categories of problem: complexity, scale, federation, heterogeneity, separation, concurrency, reliability, availability, security, and evolution. It does not include openness.

[ANSA Overview] has a subsection on distributed systems characteristics, but does not list them.

[Course Review] claims that the principles mentioned are not exactly principles. This is true; perhaps they should be described as beliefs that distributed systems pose unavoidable challenges. The principles of the ANSA architecture listed on [ANSA Overview] are far too detailed at this stage.

It is proposed that better real-world exercises [Exercises] be found to illustrate these or other characteristics of distributed systems, and this module be updated.

4.1.5 Templates for Distributed Applications (APM.1320)

[Course Review] claims that the objective of the module is unclear. It also claims that it is unclear whether it is discussing the problem statement or solution, and that it is not obvious what a template is.

The previous session (§4.1.4 *Characteristics of Open Distributed Systems (APM.1318)*) indicated that the ANSA architecture would be part of the solution. However, the templates discussed in this session are nothing to do with ANSA; they follow the GartnerGroup model. The GartnerGroup model has had wide general exposure in the technical press in the last 6 months.

The information on SIPs should be omitted; SIPs seem to be a marketing concept which has not aroused interest.

This material is a bit thin for an hour. A brief exercise would certainly help. Case study material (say from [ObjectWorld 95]) could be added.

It is proposed that this module be updated accordingly.

4.1.6 Architecture for Open Distributed Systems (APM.1323)

This module attempts to explain the ANSA/ODP meaning of architecture, and explains its benefits. It tries to do this in terms of an abstract design process; this approach doesn't really work, particularly without an example.

Potential purchasers of open systems are cynical about 'architecture'. They perceive it as an attempt to rationalize an uncoordinated set of products, or to charge more for something because it has an 'architecture'. For some commodity products, this perception may be justified.

There is a risk of ANSA being unfairly tainted by this perception.

On the other hand, those with a strong networking background will know the purpose and value of 'network architectures'. They will need convincing why another, separate, 'distributed systems architecture' is needed above it; they will also need convincing that ANSA doesn't conflict with it.

[Course Review] claims that this module has incoherent structure. This is true; we still do not have anywhere a clear explanation of what the ANSA architecture does.

[Course Review] also asserts that [Abstract and Automate] should be the basis for this module.

Unfortunately, the purpose of [Abstract and Automate] is to argue that language-based approaches using increased abstraction and automation are required. It does not explain why a distributed system architecture is needed in the first place. Nor does it give any concrete examples of how language-based approaches could solve any particular problem.

[ANSA Overview] suffers from the same problem; it explains what the ANSA Architecture is for, but not what it actually is.

It is proposed that this module be removed unless a clear explanation can be found.

Note: What should replace it?

4.1.7 Networking in Distributed Systems (APM.1331)

The aim of this module is to delineate networking architecture from distributed systems architecture; to show the benefits of each, while demonstrating that networking will not solve the problem completely.

[Course Review] says the whole idea of distributed programming environments is to avoid as much as possible the complexity of communications; communications should be transparent. Whilst a worthy idea, it is unrealistic. Because communications lies outside the ANSA architecture there has been a tendency to down-play it. It is also true that ANSA has not done much work on the technology viewpoint. In practice it is necessary to know a great deal about communications to build real systems - a point which is emphasised in this module. In the future, when applications need to control and participate in quality-of-service provision, it will be necessary to know more about communications, not less.

I believe that distributed programming environments cannot be justified solely by the complexity of communications; a stronger argument is required.

[Course Review] also claims that the link to distributed systems is missing completely and that there is no context to the talk.

The previous module §4.1.6 *Architecture for Open Distributed Systems (APM.1323)* discussed the relationship between different IT architectures, including network architectures. The link to the previous module needs strengthening, if that module is retained.

This module would benefit from a more detailed example or case study, perhaps one based on VPN (Virtual Private Network) for a telecommunications audience.

It is proposed that this module be clarified, and a detailed example included.

4.2 Understanding Distributed Systems Architecture - Day 2

4.2.1 Review of Day 1

This is not a proper review session. It simply consists of slides selected by the presenter from the previous day; typically 10 or slides. Currently it only lasts 5 minutes. A proper review session would depend on the exact contents of the previous day, which is why it has never been written.

It would be best supported by a brief 10-minute exercise to warm people up.

It is proposed that the format of these review sessions be discussed further.

4.2.2 Introduction to CORBA and DCE (APM.1322)

This module compares and contrasts OMG and OSF, and also the basic approach of CORBA and DCE. It does not attempt a description of CORBA or DCE.

The idea is clarify their respective problem domains, strategies, and marketing approaches. It is possible that 'compare and contrast' is not the right way to tackle this. [Course Review] says that "I cannot understand why this would interest people at this level of detail".

[Course Review] also notes that the relationship between (generic) architecture and (specific) distributed processing environments (CORBA, DCE) is not clearly explained.

The idea of Remote Procedure Call (RPC) should be introduced in this module.

It is proposed that the aims of this module be clarified, and the content be updated to the latest state of the market (including non-confidential information from [ObjectWorld 95] and [Otelso Workshop]).

Note: There is also a need for a variant of this module that covers CORBA alone. This might be better based on the standard introductory presentation from OMG.

4.2.3 The ODP Reference Model (APM.1336)

[Course Review] questions whether this module is needed at all.

It is extremely difficult to promote ODP in the absence of market awareness, a method of use, or delivered products. [Course Review] says that the module does not explain the benefits or a method of use. No comprehensive method of use for ODP has been defined (by anyone); the first steps being in MCI and ECP.

Unfortunately, the existence of ODP is a justification for promoting the ANSA architecture; ANSA is non-proprietary because it is so close to ODP. If we down-play ODP, we need to explain why the Computational Model and Engineering Model are important.

This may be best achieved by working through describing an example system from each viewpoint (say the SimpleBank). Unfortunately there is no existing example we can use; for telecommunications companies, perhaps an example from TINA, ICODP, the [Otelso Workshop], or ROSA would be best. (Note that we have had particular difficulty with explaining Enterprise modelling in the past.)

Note: Telecommunications companies (BT, CNET) have asked for ODP content in this course. It may be appropriate to revise it and retain it as an optional module.

It is proposed that this module be removed, nonetheless; passing mention of ODP can be made elsewhere. It is a "secret we keep to ourselves". Note that references to ODP in other modules will need to be replaced.

4.2.4 Management of Distributed Networks (APM.1324)

As originally scoped, this module tries to cover too much ground. It covers several open frameworks for:

- network management (mainly hardware components)
- system management (mainly hardware and software components)

It also tries to cover:

- applications management (i.e management *of* applications software)
- management applications (i.e. applications that manage networks, systems, and other applications)

It does not cover management of services.

[Course Review] notes that this module is somewhat out of sequence in the course; there is a brief reference to ODP (the previous session, §4.2.3 *The ODP Reference Model (APM.1336)*), but this does not make a strong case for ODP being a preferred architecture for management.

Use of RPC as a preferred technique and OO for encapsulation is mentioned, but this hardly seems enough for a 1-hour module.

Although ANSA Phase 3 has been focused on the vision of a Management Engine, we do not yet have results that can be applied to current management problems. Work in ISA/DEMON (on monitoring) and DOMAINS (on policies) has not been carried forward. We have also not addressed the legacy issues with management protocols (notably SNMP and CMIP).

A great deal of research has been done on management in recent years (e.g. in SysMan [ObjectWorld 95]); it is to be determined how valid and useful this is.

Course feedback indicates that application and service management is of great interest. The failure of OSF DME has left a great gap, and people are unsure where to turn.

Note that this presentation is in PowerPoint format, and should be converted to FrameMaker.

The proposal for this module is for further discussion.

4.2.5 Distributed Communications Techniques (APM.1340)

[Course Review] claims to fail to see the message or the need for this module, and says that it should be dropped or seriously revised.

This module is aimed at the system architect, who may be aware that different kinds of middleware products and strategies are available, but is unsure of the strengths and weaknesses of each.

This module concentrates on RQM (Robust Queued Messaging), also known as MOM (Message Oriented Middleware). It aims to prove that an RPC interface to RQM systems is a viable and useful approach.

Such an interface now exists; the CORBA Event service [CORBAservices]. It might be better to base this module on the CORBA Event service.

Details of implementations of the CORBA Event service should be included when products are released. Information on other RQM products is now out-of-date and should be updated.

This module is also weakened by the absence of specific examples. The Prussian Generals problem is not a good example; it doesn't link with the rest of the module and should be removed.

The detail on RDA should be moved to the following session.

It is proposed that this module be based on the CORBA Event service. (Note that this introduces a dependency on a prior discussion of CORBA Object Services, which are currently covered in a later course.)

4.2.6 Distributed Database and Distributed Systems (APM.1332)

This module is concerned with the unification of RPC and RDA techniques. Unification of these techniques is a specialist topic.

A replacement module (APM.1461) covering RDA and Distributed Database in detail has already been written. This topic is important to mainstream client/

server systems, but is not one that in which APM have much expertise. However, it is felt to be essential.

It is proposed that the replacement module be reviewed for use in this course.

4.2.7 The Computational Model (APM.1327)

This module gives a brief overview of the ANSA/ODP Computational Model, including the object interaction diagrams.

It is worth considering whether the Computational Model per se is directly relevant to applications programmers, if we are not emphasising ODP. The Computational Model is really relevant when contrasted with the Engineering Model (which is not part of this course, and not relevant to many applications programmers anyway). Just as with ODP, the Computational Model is a “secret we keep to ourselves”; its importance is the influence on the CORBA architecture and specifications, but the Computational Model itself is not relevant here.

Most applications programmers will only be concerned with operation invocations using implicit bindings. (Streams and explicit bindings can be dealt with in other modules.)

Note: CNET have said that they do require brief coverage of streams and signals in this module.

The easiest way to deal with this topic is to treat it as the convergence of RPC and Objects. [Course Review] suggests using the §4.4.2 *Objects in Distributed Systems (APM.1350)* module from “Building Applications with Distributed Objects” instead. This means explaining Objects, before explaining the stricter rules and other implications of Distributed Objects (strict encapsulation and conformance).

It is proposed that this module be removed and detail be merged with the §4.4.2 *Objects in Distributed Systems (APM.1350)* module, suitably reworked (see below). A reference in passing to ‘the Computational Model’ needs to be included, to preserve the integrity of other modules. (Note that this means that a substitute module is required in “Building Applications with Distributed Objects”.)

4.3 Understanding Distributed Systems Architecture - Day 3

4.3.1 Review of Day 2

Same comments as corresponding session on previous day (§4.2.1 *Review of Day 1*).

4.3.2 Trading and Federation (APM.1330)

This module has been a success when examples have sparked off discussion.

We have considered using the Simple Bank example here as an example of trading. Unfortunately, it is an unrealistic example - real ATM/banking systems would not use ANSA/ODP trading in this way.

Course feedback has also shown some scepticism that trading is relevant to mainstream distributed systems. (It is promoted in ICL’s DAIS, but in a fairly low-key way. It has also taken a long time for OMG to consider it for a CORBA Object Services.)

[Course Review] says that treating trading and federation together is inappropriate. In the past, ANSA documents have often covered trading and federation in one breath. There is no strong reason to do so; federation could equally well be covered along with security, for instance.

In this module, federation is introduced by the need to distribute the trading service itself, because in the future there will be millions [Course Review] says “scaling is not the reason why we have federated systems.” However, it is one of the reasons. We have promoted federation as a means of crossing controlled boundaries. Right now, many organizations will be satisfied with crossing boundaries anyhow; they will dispense with the control. Any interconnection is better than none. Our forward-looking view of federation may be seen to over-ambitious (even though we can demonstrate it working!).

Note the this module depends on the diagrams introduced in the previous session, §4.2.7 *The Computational Model (APM.1327)*. Therefore the notation at least of the Computational Model must be covered here. If the module §4.4.3 *Specifying Services for Distributed Systems (APM.1348)* is also transferred to this course, this will show how object references can be passed as parameters, and the CORBA IDL for the [ODP Trader] can be used. (We can continue to point to the ANSAware Trader as proof-of-concept.)

The [ODP Trader] has now reached DIS; this module should be aligned with it. Material from [Trader Presentation] should also be included.

It is proposed to update this module to include a worked example of trading in a plausible application; such an application may be difficult to find. The discussion of federation should be retained.

4.3.3 Service Quality in Distributed Systems (APM.1321)

This module confounds two separate ideas:

- Service Quality in the sense of business service level agreements (SLAs) and Total Quality Management (TQM) - enterprise issues
- Quality of Service (QoS) in the sense of communications parameters - technology and engineering issues

The ANSA Phase 3 work on federation, and also on trading does show how a common ODP framework can embrace both. However, even standards on QoS are some way in the future, and this is best regarded as a current research topic.

This module does form a natural continuation of §4.3.2 *Trading and Federation (APM.1330)*.

Real examples would be better than the initial exercise on best effort and guaranteed service (even though it is fun).

[Course Review] notes that some detail is very confusing in later slides, and the order of slides is not clear.

It is proposed to remove this module (need to check for other modules that depend on it.)

Note: What to replace it by?

4.3.4 Distributed Workflow Applications (APM.1319)

General course feedback is that this module does not belong in the course. There have been comments that this is a business process automation issue, rather than a distributed systems issue.

This is a pity; the aim of this module was to show that distributed system techniques can bring benefits not only to specific applications, but also to generic classes of applications - in this case, workflow applications.

If this module is removed, there is very little focus on applications in the remaining modules. No other generic class of applications seems appropriate on which to base a new module.

This module is in two parts; the first part on business process issues is not well linked with the second part on event mechanisms (which only get brief coverage anyway).

It is proposed, nonetheless, to remove this module, due to the consistent feedback.

Note: What to replace it by?

4.3.5 Security in Distributed Systems (APM.1334)

This module has had polarized feedback; some participants think it entirely irrelevant, many enjoy it.

Security is a topic where ANSA have specialist knowledge. I do have concern that the object-based view of security promoted by ANSA is a minority view in the wider marketplace, and is unlikely to appear in commercial off-the-shelf products in the foreseeable future. The presentation here is one-sided, and may mislead.

[Course Review] also notes that slides 40 and beyond should be in an appendix; this is agreed.

Note that this module is in PowerPoint format and will need to be converted. (Some of the diagrams are complicated.)

It is proposed to include some balancing material on other security techniques already being used in distributed systems (e.g. Kerberos and DCE), plus information on the CORBA Security Service. (Note that the CORBA Security Service has not yet been finalized.) This will also require product knowledge.

4.3.6 Services in the Electronic Marketplace (APM.1326)

[Course Review] says this might fit in better earlier in the course.

The idea of this module was that it was a 'Vision Of The Future', which is usually the last session in the course. It also aimed to show that Security, Service Quality, and Trading are essential to services in the electronic marketplace; however, [Course Review] says that "there is no good tie in to what was discussed in the 2 and half days before."

It could be that the 'Vision of the Future' is too far distant, or implausible.

It is not clear whether the arguments for Security, Service Quality, and Trading really are compelling enough for this tie-in.

[Course Review] likes the examples concerning remuneration. I am concerned that remuneration is irrelevant to most MIS/IT organizations which do not require this strong a notion of billing for internal services. Furthermore, there

are imminent products providing (limited) support for remuneration; the ANSA Phase 3 work has been overtaken by events.

Now that WWW technology has a very high profile in the mass media, it may be appropriate to say more about it here. There are bound to be cynics, but I feel we have to be able to come up with a business case for WWW technology for internal dissemination of information within an organization.

[Course Review] also says that this sessions has too many slides; this is agreed.

It is proposed to remove this module, but retain it as an option. When required, this module should be updated to reinforce the Security, Service Quality, and Trading, to reduce the material on remuneration, and add more material on WWW.

4.3.7 Course Roundup (APM.1335)

This is very weak; it is just copies of significant individual slides from previous sessions.

This module needs the same treatment as §4.1.2 *Introduction to Distributed Systems (APM.1328)*. However, it should be a brief and cogent summary and recap - but without simply repeating what was said earlier. There is no existing ANSA presentation to draw on here.

I feel that the slides on Learning More are useful, but there has been no specific feedback to confirm this.

It is proposed that recap slides this module be rewritten. It is vital that course participants leave with a positive message ringing in their ears!

4.4 “Building Applications with Distributed Objects” - Day 1

4.4.1 Welcome

Same comments as for same session in previous course (§4.1.1 *Welcome (APM.1339)*).

4.4.2 Objects in Distributed Systems (APM.1350)

This module attempts to explain what objects are (for people who are unsure). It must then explain how distributed objects differ from objects in programming languages.

This is difficult because we have to explain the concepts, then explain why one of them (inheritance) doesn't really matter.

The main concepts (polymorphism, encapsulation, inheritance) need more time spent on them - in examples, rather than definitions. The Shapes example is a start, but needs more work to make it plausible. (It also needs a correctly worked answer.)

It may be preferable to use some more standard object examples (e.g. Vehicles from various presentations in [ObjectWorld 95].)

It would be easier to start from C++ (classes) or the SQL3 object model?

I think this may be too many topics for one hour.

The distributed object technologies (OLE/COM, OpenDoc) are not a good fit to the definitional approach in the rest of this session.

The analysis in [Course Review] gives a sample story-board, for this session.

It is proposed that this module be further discussed

4.4.3 Specifying Services for Distributed Systems (APM.1348)

This module is an introduction to IDL, specifically, CORBA IDL.

Some participants who do not have recent programming experience may need to take this slowly; data types such as discriminated unions and enumerations may be new to them. This will be tricky when giving this session to participants with mixed experience.

It might be helpful to show the standard C++ CORBA mappings for each construct.

This session needs plausible small examples, concentrating on semantics rather than syntax. (Most programming language manuals are very bad on this - including the CORBA specifications. It may be best to use the Simple Bank example here. (Perhaps skim through some standard C++ reference books for brief supplementary examples.)

It is not clear whether the comparisons with ANSAware IDL and DCE IDL are helpful.

The analysis in [Course Review] gives a sample story-board, for this session.

It is proposed that this module be updated to include more examples.

Note: Which examples?

4.4.4 Remote Procedure Call in Distributed Systems (APM.1344)

This module rather labours its points.

This module might be better if it explained how RPC actually works (as presented in [Power Programming with RPC]). It might also be better if presented as in the DCE manuals. It might be better if this module focused on one particular technology (DCE or CORBA). Unfortunately the mechanisms for manipulating the RPC itself in CORBA are typically vendor-specific (for example, as in Orbix).

The coverage of idempotence is confusing. Idempotence is not a CORBA concept, and has a different shade of meaning in DCE.

A worked example showing the flow of control, particularly of the use of asynchronous RPC, would help.

The material on multi-threading and pipeline models might be better merged with §4.5.1 *Concurrency in Distributed Systems (APM.1365)*. This is especially so, since explicit multi-threading is typically used in servers rather than clients.

It is proposed that this module be further discussed.

4.4.5 The Engineering Model (APM.1351)

Same comments as for §4.2.7 *The Computational Model (APM.1327)*.

The difficulty here is that ANSAware is the only infrastructure that has been built using the ODP Engineering model concepts and terminology. It is hard to relate these concepts and terminology to CORBA products.

It is proposed that this session should be removed. The engineering concepts of capsules, clusters, and so on should be merged into an example exercise.

Note: What to replace it?

Note: CNET have requested that this module be retained, precisely so as to relate the Engineering Model to CORBA concepts.

4.4.6 DCE Distributed Services (APM.1373)

This module covers the ground, but is rather superficial.

The material on the DCE Directory services must be tied together with a specific application example.

We now have access to detailed [DCE Internals] course notes; the technical detail must be checked and brought up to date with DCE 1.1

Finally, a plausible example that uses all these services should be presented.

It is proposed that this module be updated.

4.4.7 The CORBA Object Management Architecture (APM.1345)

This module covers the Core Object Model, IDL/SII versus DII, ORB implementations and ORB integration. The first half of this module is a recap of CORBA IDL. This is rather unnecessary, given the detail in §4.4.3 *Specifying Services for Distributed Systems (APM.1348)*.

This module is currently a rather uneasy balance between high-level summary and low-level detail. It covers too many concepts for a single module. Furthermore, there are no examples given. It might be better to compare two ORB implementations (or two applications?) and show how they differ. This might in turn conflict with §4.6.1 *CORBA in the Real World (APM.1379)*.

Alternatively, it might be better to split out client-side issues from server-side issues. and cover these in separate modules.

It might be helpful to tie this more closely to ODP ideas.

This module needs also updating to ORB 2.0 (e.g. for the DSI); and an indication of the ORB Portability issues that are not yet resolved must be included.

Note: It is not yet clear which OMG documents define CORBA 2.0

It is proposed that this module be further discussed.

4.5 “Building Applications with Distributed Objects” - Day 2

4.5.1 Concurrency in Distributed Systems (APM.1365)

This module covers concepts of concurrency and locking.

Multi-threading needs a more careful explanation before moving on to locking constructs. There should also be some coverage of optimistic concurrency control techniques as they are used by telecommunications systems, and become more relevant in distributed systems.

This session is based on the constructs of ANSAware, which uses non-standard concurrency mechanisms. It should be based on the CORBA Object Concurrency and CORBA Transaction services (both now published in [CORBAservices]).

Note that CORBA does not specify any multi-threading interfaces. If the discussion of multi-threading is transferred from §4.4.4 *Remote Procedure Call in Distributed Systems (APM.1344)*, a brief discussion is required.

Once again, needs a good application example (possibly Simple Bank).

It is proposed that this module be further discussed.

4.5.2 Building Applications with ANSA (APM.1366)

This module was include to give some background to ANSAware, in response to feedback from ANSAware users. It now includes rather too much coverage of ANSAware detail.

The reasoning behind preferring Language Extensions over APIs needs strengthening with fact and example; at the moment the argument simply isn't made strongly enough. Some people believe in the Abstract part of [Abstract and Automate]; but not the Automate part, preferring C++ class-based solutions. It is also unclear how this relates to the CORBA language mappings, in particular the C++ mapping. This module should be aligned with ANSAware/RT (if there are any differences).

If this approach to Language Extensions is to be reinforced, the example must include some ANSA PREPC samples.

If not, this session should concentrate on the Factory and Node Manager. Note that the CORBA Lifecycle service corresponds to the Factory service; there is no equivalent to the Node Manager. The Factory and Node Manager services need a more detailed description of their exact functions.

It is proposed that this module be further discussed.

4.5.3 Naming in Distributed Systems (APM.1343)

This module is too theoretical. A couple of key examples should be extracted and merged with the §4.4.6 *DCE Distributed Services (APM.1373)* or §4.5.6 *CORBA Object Services (APM.1349)* sessions.

It is proposed that this module should then be removed.

4.5.4 Dependability in Open Distributed Systems (APM.1341)

This module describes the basic ideas of dependability. However it is rather theoretical and does not explain how to build such systems. The important theoretical ideas are passed over too rapidly. It does not apply any CORBA, ODP, or DCE concepts to dependability. The basic ideas should be merged with §4.6.2 *Replication Techniques for Distributed Systems (APM.1358)*

It is proposed that this session should then be removed.

4.5.5 Designing Applications with CORBA (APM.1352)

This module begins by looking at the ORB Object and Interface Repository interfaces. These are not good examples of IDL; they are very abstract, and many applications will not need to use these interfaces.

There is then a very brief discussion of the Basic Object Adaptor (BOA) and activation policies.

This module then concludes with a practical session in designing a simple application - the Daemon Game. This needs a worked answer. It is also not a very good example. Perhaps the CORBA card game (see [Exercises]) would be a better choice. It might be a better idea to make the practical session a module in its own right.

It is proposed that the practical session be a module in its own right; the other material should be transferred to §4.4.3 *Specifying Services for Distributed Systems (APM.1348)*.

4.5.6 CORBA Object Services (APM.1349)

This module is too superficial. It is too ambitious to expect to cover three services in one session, so ends up with too many slides. The object services described are the simply those that were first specified. Once again, this module needs examples to show why and how you would use these services.

This module needs updating to the latest [CORBAServices].

There are two choices here: a very brief summary of every service (appropriately grouped), or go into a selected few in detail. (Maybe need two alternative presentations here.)

It is proposed that this module be replaced by a brief summary of every CORBA service, explaining how and why you would use them, and when vendor implementations are likely to become available.

4.6 “Building Applications with Distributed Objects” - Day 3

4.6.1 CORBA in the Real World (APM.1379)

This module is confidential to the ANSA consortium.

It consists of a brief overview of the Simple Bank example (in CORBA IDL), followed by summaries of each ORB evaluation that we carried out.

The Simple Bank example is not a good fit with the rest of the module; it would be more useful as a general example, for instance in §4.4.3 *Specifying Services for Distributed Systems (APM.1348)*

It might be worth commenting on the vendor-specific extensions (e.g. Orbix's smart proxies, filters, loaders, and locators).

It might also be worth commenting on each vendor's approach to interoperability.

It is proposed that the Simple Bank example be transferred to §4.4.3 *Specifying Services for Distributed Systems (APM.1348)*, and this module shortened accordingly (perhaps to half-an-hour). (It can be lengthened again as more information is available from the ANSA Phase 3 ORB evaluations.)

4.6.2 Replication Techniques for Distributed Systems (APM.1358)

This module concentrates on ANSAware and ISIS, but from a theoretical point of view. The work done in ANSAware on groups is now out-of-date, and has not been carried forward into ANSAware/RT (GEX is no longer supported). Some

of the ANSA sponsors are already using Orbix+ISIS; we lack even basic knowledge about this product.

This module concentrates on engineering issues and concepts, rather than the application view. Once again, this module lacks any application examples.

There has been a lot of interest in database replication recently, and a fair amount of technical detail has been published about the various approaches taken to replication in Oracle, Sybase, SQL Server, and so on. Reports from early adopters are beginning to appear. Although database is not our specialist area, it may be worth refocusing this session on databases (at least partially) and providing a critique with respect to transparencies (since database vendors have adopted the transparency terminology here).

It is proposed that this module be further discussed.

4.6.3 Exploiting High Performance Networks (APM.1357)

This module introduces some basic ideas of LAN interconnection via a WAN, and then discusses the impact that this will have on distributed applications, particularly the need for end-to-end quality of service support

The introductory part on LAN bridging via a WAN does not fit well.

If this module belongs at all, it should be in a specialist course on distributed multimedia (or perhaps telecommunications). Standards and products are only just beginning to emerge, and will be a long while before distributed applications are running over these networks.

Much of the material here is better covered by [Total Area Networking].

It is proposed that this module be removed.

4.6.4 Real-time Distributed Systems (APM.1353)

This module discusses the basic concepts of real-time systems.

Now that ANSAware/RT is available, it might be preferable to base this module on that. There is already a presentation given at ANSAworks '95 ([ANSAware/RT]), with detailed handwritten notes. If so, this module should be written so that it is not confidential to the ANSA consortium.

However, it seems that most people are interested in soft real-time distributed computing; they are concerned with dimensioning, benchmarking, and performance engineering, rather than guarantees.

IONA have a real-time product (Orbix+VxWorks). This product should be mentioned; it may be useful to gather more technical information about it.

It is proposed that this module be further discussed.

4.6.5 Multi-media in Distributed Systems (APM.1354)

This module attempts to tie together the previous sessions on real-time, high-performance networks, and replication. However, the overall effect is too specialised and superficial.

The material on binding is currently only relevant to research projects. There are no products yet which use this binding model; DCE doesn't, and neither does CORBA. The material from this module would be more relevant to telecommunications, and should be reused in a new module with a telecommunications focus, when required.

It is proposed that this module be removed.

4.6.6 Course Round-up (APM.1356)

Same comments as for same session in previous course (§4.3.7 *Course Roundup (APM.1335)*).

5 Course presentation improvement

5.1 Session length

Several course participants commented that 1-hour sessions were too long. (University lectures tend to be just under an hour, school lessons typically 50 minutes.)

I have noted that the session timings varied much more widely on the BT courses than on the public courses given in 1994. This may be because the public courses had fewer participants, and that with a smaller group it is easier to stimulate discussion to compensate for lack of questions. Time control would be easier if there were more exercises, and less speaker presentation.

Speaker notes should include timings at least every 15 minutes.

With the existing courses (and course presenter), under-running happens more often than over-running. This may be because of speaking too fast, insufficient material, or just generally poor time control.

Some course participants would have preferred longer breaks.

Note: Recommend that we standardize on 45 minutes, including questions?

5.2 Discussion sessions

While the discussions were useful, they need tight control to avoid some participants losing interest. Some of the planned discussions sessions are not sufficiently illuminating to justify the time spent; these should be cut back.

5.3 Slides

5.3.1 Slides quality

All slides are black-and-white. As stated in [Training Plan], this is sufficient for 'good' quality. 'Better' quality would require colour slides (as for TC presentations). 'Best' quality would require a projection panel, using PowerPoint.

Feedback indicates that this policy is acceptable (at least for 1995), and that a projection panel is not yet required generally.

The issue of conversion between FrameMaker and PowerPoint remains problematic, and is discussed elsewhere.

5.3.2 Time and content per slide

There is agreement that between 2 and 3 minutes per slide is ideal. However, some presentations include many more than this. Often this is because several slides of text had been used where a diagram would be more appropriate (but

no standard diagram exists). Often it is because the slides include text that would be better in audience notes. Sometimes it is because the content of the slides is thin.

A good diagram is preferable to text, but diagrams are time-consuming to prepare. Existing material will be changed only where a compelling new diagram is already conceived.

There is a genuine difference of opinion about how much information should be included per slide. Our training courses slides are broadly consistent with [DCE Internals] in that respect; ANSA Technical Committee presentations often contain 9 or 10 lines of text, and pack in more information per slide.

No change is proposed to the density of existing material.

To help keep course running to time, the speaker notes for each module should include time checkpoints every few slides.

5.4 Course handouts

Several course participants felt that the course handouts were inadequate, and that it would not be possible to understand the slides when rereading them at a later date.

The proposed solution is that the course handouts should include audience notes interleaved with copies of the slides. This is consistent with most public training courses, but is a step beyond any previous APM presentation material. (ANSA Technical Committee presentations do not include audience notes; presumably this because they are presented once only, and supporting ANSA documents are the permanent record of what was said.)

An experimental FrameMaker template has been generated; including this template in the APM document management system, and adding an option to print only even-numbered/odd-numbered pages is approximately 2 days work.

Reformatting an existing presentation to include audience notes pages takes about 1 hour. Writing the audience notes themselves would take approximately half a day per presentation; not all slides need audience notes., This may involve transferring some material from the slides to the audience notes, particularly slides consisting entirely of text - this may address the concerns about the number of slides (see above.)

Following this, the presentation must be reviewed again (at least proofread), then masters of the slides, the handouts, and speaker notes reprinted; this is about 1 day.

For consistency, this must be done for all modules in a course (or for none of them). Therefore, for a 3-day course involving 18 modules, this would be roughly 25 days effort. Where modules are shared between courses, the job need only be done once.

There is a problem with the presentations written in PowerPoint. PowerPoint has only one notes page and cannot support both speaker notes and audience notes. The proposed solution is to convert the existing speaker notes to audience notes; it is possible to fudge speaker notes by putting them outside the normally printable area. However, most if not all of the PowerPoint presentations will be converted to FrameMaker before handover to CNET, anyway.

Presentations with audience notes will make it easier for course presenters to understand the material, and therefore make the handover of course material to CNET more straightforward. (It would increase translation costs, since there will be more text to translate, but this cost will not be borne by APM.)

It is therefore proposed that upgrading existing presentations to include audience notes be done as part of the CNET training project [CNET Proposal]. (If this were not done, other background material would be needed for CNET in any case.)

Experiments have been made to produce a Table of Contents based on the title of each slide. This can be done manually using FrameMaker, but is impractical using the APM document management system, since it would span multiple documents. It could not be done with PowerPoint presentations.

Indexing with FrameMaker has to be done manually by tagging each entry. This would be much too time-consuming.

The course handouts include some ANSA reference material. This could be enhanced by including a wider range of reprints, but this would increase costs. This is for further discussion.

5.5 Course assessment questionnaire

[Course Review] suggests that more specific information is required from course assessments, for example per-module feedback.

It is hard to get course participants to fill in more than 2 pages, unfortunately. No change is proposed, other than fine-tuning.

5.6 Presenter preparation

Presenters must be confident with

- target audience: can I handle this type of audience?
- quality of material
- type of subject matter: e.g. product-related, theory, etc.
- belief in the story

This requires briefing notes for the course material. This is for further review.

6 Course development process improvements

The analysis in [Course Review] suggests a detailed top-down approach to the development and review of course content, based on the approach taken for ANSA Phase 3 technical documents. This analysis is valuable, but a few points are worth emphasising:

- unlike many of the ANSA Phase 3 technical documents, the course content must be expected to evolve
- the analysis assumes that the course material is being developed from scratch

These comments are not course-specific.

6.1 Aims, objectives, and learning outcomes (AOLO)

Aims, objectives, and learning outcomes are the explicit reasons for participating in the course. However, it is worth bearing in mind that each person will also have their own specific business motives (increased profit, decreased risk etc.) and personal motives for being on the course (a break from the daily routine).

6.1.1 Aims

Aims specify the ultimate purpose of the course/session/activity. An explicit statement of aims allows participants to determine whether the aims match their motives.

Aims may be vague. Aims may extend beyond the time and nature of the course/session/activity; it is important to make this clear so that participants have realistic expectations.

Aims correspond to a business vision.

6.1.2 Objectives

Objectives specify precise observable and measurable actions that participants will carry out during the course/session/activity in support of the aims.

Objectives are usually stated in the form:

“By the end of this course/session/activity, participants will have ...”

Objectives are the basis for evaluation and assessment. Associated with each objective are:

- performance criteria for participants
- assessment method, conditions, and evidence
- range indicators

Objectives correspond to a business mission.

6.1.3 Learning outcomes

Learning outcomes specify precise observable, measurable, long-term, and permanent changes in participants' approaches, knowledge, or skills.

Learning outcomes are usually stated in the form:

“As a result of this course/session/activity, participants will be able to...”

Learning outcomes correspond to business goals.

Roughly speaking, the participants' achievement of objectives is assessed by “continuous assessment”. The participants' achievement of learning outcomes is assessed by “final examination”.

6.1.4 Proposal

It is proposed that the aims and objectives of each course and module be specified. (It is difficult to see how learning outcomes can be specified for APM courses, particularly since we are not in a position to assess them.)

6.2 Outlining and story-boarding

It was observed in many of the modules, the ‘story’ of the session was unclear, even though the objectives of the session are stated, and conclusions are drawn.

A clear ‘story’ is vital for the course presenter. It is also important for course participants when re-reading the course material later (referring to copies of the slides).

The lack of a clear ‘story’ may be due to:

- unclear aims and objectives; too few objectives, and these objectives not being clearly related to the aims
- lack of continuity between modules within the course
- problem and solution being intermingled. If a problem is described, a solution may or may not already be available
- past and future being intermingled
- weak conclusions
- lack of audience notes

It is proposed that outlining and story-boarding techniques may help.

6.2.1 Strength of argument

The business case and technical case in a module, considered as a logical argument, must be:

- correct: facts must be accurate and honest and the argument must be free of logical flaws
- comprehensible: premises must be understandable to target audience and new terminology and flow of argument must be straightforward and clear; there must be a clear separation of problem and solution
- convincing: premises must be plausible, and the argument must be well-supported by evidence and proof

- **compelling:** premises and conclusions must be relevant and important to the target audience. It also helps if the conclusions are exciting and palatable both to APM and to the target audience

If the story-board is mainly factual, this is straightforward.

6.2.1.1 *Correctness*

Correctness is not a significant problem with APM presentations. However, the remaining issues are.

6.2.1.2 *Comprehensibility*

Many of the sessions are not comprehensible enough. They often require knowledge of special concepts or terminology that the target audience does not have. It may be inappropriate to introduce these concepts solely for explication; it would be better to change to avoid them by changing the argument to use concepts familiar to the target audience, even if they are different from those traditionally used in ANSA. Often the argument falls back on abstraction, which can be difficult to follow. (This is a general problem with material derived from ANSA Architecture Reports and Technical Reports.)

6.2.1.3 *Conviction*

Many of the sessions are not convincing enough. Either the evidence is weak (few case studies and examples), or the continuity of the argument is genuinely thin. This may be because we have not refined our own arguments, (the ANSA party line is weak), or because it is not an ANSA-specific argument (so we are relying on other people's weak arguments).

6.2.1.4 *Compulsion*

Many of the sessions are not compelling enough. Either the premises are weak ('that doesn't apply to us'), or the conclusions are weak ('so what?'). It isn't enough prove we have a solution; we have to demonstrate that it is the best solution, and that solving the problem will yield better returns than investing the money and effort in solving some other business problem.

Note that the premises and conclusions of the argument don't have to match the aims and objectives of the module (although it would be ideal if they did).

The summary should consist of, in order of preference:

- **recommendations:** calls to action, do's and don'ts
- **conclusions**
- **facts:** restatement of key facts

6.2.2 **Outlining**

An outline is a logical breakdown of the topics covered, in the form of a hierarchical list or mind-map. (The road-maps used in §4.1.1 *Welcome (APM.1339)* are outlines. An outline shows how the topics are related, but it gives no idea of their relative importance. The outline should be immediately comprehensible to the course participants.

Inferring the course from the existing slides can be difficult.

6.2.3 Story-board

Note: A standard textbook reference for story-boarding would be helpful here

A story-board is a sequence of a key-frames. A key-frame is a diagram, example, or statement which is fundamental to the presentation. Unlike an outline, a story-board is in sequence; also the story-board does not have to cover all the topics.

The story-board is an aid to the presenter. It is not shown to the course participants.

A key-frame does not have to be a diagram. Diagrams are generally preferred, but can be misleading. They can also be rejected by the participants if they are unfamiliar. If there is a sequence of diagrams, it should be possible to infer or recollect the argument from that sequence of diagrams.

It is recommended that there should be one key-frame for the premises, and another for the conclusions.

A story-board should reflect the approach of the presentation; if the presentation is historical (past -> present -> future), the story-board should reflect this.

Different audiences may require different stories (even if the premises and conclusions are the same). This makes it difficult to produce generic presentations.

6.2.4 Proposal

It is therefore proposed that:

- when specifying a module, first state the individual premises and the conclusions clearly, and check they are relevant and important to the target audience (the proof should come later). This should become the business case and technical case in the Abstract. It should be possible to explain these face-to-face in less than a minute
- the Abstract on the document cover page should include the explicit logical argument
- somewhere the aims and objectives should be explicitly stated (in the Abstract?)
- somewhere there should be an outline of the topics of the presentation (possibly in the form of a mind-map)
- the 'story' should include 'key-frames' (important diagrams and at least one example)
- the story should include 'checkpoints' for the presenter to confirm understanding and review what has been said

For existing presentations, if they are to be retained:

- the course summary for the presenter should include a story-board and outline for each presentation
- the Abstract for the presentation should updated to include the aim and objectives, the premises and conclusions, and the business problem, technical problem and solution.

Updating the original summaries in [Modules] is to be determined.

6.3 Course development teams

The following issues arose when delegating course development:

- lack of proper presentation templates and documentation for them
- lack of support for workflow for the course development process (and documented procedures for it, particularly where it differs from the ANSA process)
- control of time (when people are working on other things), both the writing and the review process
- lack of top-down documentation for how the course material fits together (i.e. documentation of the course content)

It is proposed to document these procedures.

7 Course structure

7.1 Courses

This table includes all the courses mentioned in our training brochure

Table 7.1: Offered course titles and code

Title	Short code	Long code	Days	Offered by
The Impact of Distributed Processing		IDP	1	Hoskyns
Introducing Distributed Systems to the Enterprise, Introducing Distributed Processing to your Enterprise	IE	IDPE2, IDPE	2	APM, Hoskyns, Peritas
Introduction to Distributed Systems Architecture	IA	IDSA2	2	APM, Peritas
Understanding Distributed Systems Architecture	UA	UDSA3 DSA	3	APM, Hoskyns
Tutorial on Distributed Architecture and Standards	TA	TDAS1	1	APM
Building Applications with Distributed Objects	BD	BADO3 BADO	3	APM, Hoskyns, Peritas
Writing Distributed Applications using ANSAware	AA	WDAA2	2	APM, Peritas
ANSAware In Depth	AD	AD2	2	APM, Peritas
Developing Manageable Distributed Applications	DA	DMDA3	3	-
Designing and Engineering Distributed Systems	DS	DEDS2	2	-

Courses AA and AD are for further discussion. Courses DA and DS are mentioned in our training brochure, but we have no plans to develop them.

7.2 Course modules

The following table is for further review:

Table 1: Course modules by course

APM	Module	TA/ TDAS1 [DSAS for CNET]	[IDP]	IE/ IDPE2	IA/ IDSA2 [IDDS for CNET]	UA/ UDSA3	ADDS for CNET	BD/ BAD03
1471	<i>Welcome [to Impact of Distributed Processing]</i>		1/0					
1473	The Role of Information Systems		1/1					
1477	Elements of Client/Server Computing		1/2					
1472	Exploiting IT Trends in the 1990s		1/3					

Table 1: Course modules by course

APM	Module	TA/ TDAS1 [DSAS for CNET]	[IDP]	IE/ IDPE2	IA/ IDSA2 [IDDS for CNET]	UA/ UDSA3	ADDS for CNET	BD/ BAD03
1478 ^a	Object Technology for Information Systems		1/4					
1329	<i>Welcome [to Understanding Distributed Systems Architecture]</i>					1/0		
1328	Introduction to Distributed Systems	1/1 ^b				1/1		
1395	Introduction to Distributed Systems				1/1			
1317	Distributed and Networked Operating Systems				1/2	1/2		
1318	Characteristics of Open Distributed Systems					1/3		
1320	Templates for Distributed Applications		1/5		1/3	1/4		
1323	Architecture for Open Distributed Systems					1/5		
1331	Networking in Distributed Systems					1/6		
-	<i>Review of Day 1</i>					2/0		
1322	Introduction to CORBA and DCE	1/2	1/6			2/1		
1336	The ODP Reference Model	1/3				2/2		
M1324	Management of Distributed Networks				2/4	2/3		
1340	Distributed Communications Techniques					2/4		
1332	Distributed Databases and Distributed Systems					2/5		
1327	The Computational Model				1/4	2/6		
-	Structuring Distributed Applications				1/5			
1468	Distributed Client/Server in Action		1/7					
1475	Course Roundup and Action Plan		1/8					
-	<i>Review of Day 2</i>					3/0		
1330	Trading and Federation				1/6	3/1		
1321	Service Quality in Distributed Systems					3/2	1/2	
1319	Distributed Workflow Applications					3/3		
1334	Security in Distributed Systems					3/4		
1326	Services in the Electronic Marketplace					3/5		
1335	Course Roundup [for UDSA]					3/6		
-	<i>Welcome [to ADDS]</i>						1/0	
-	Distributed Systems Update						1/1	
1355/ 1375	<i>Welcome [to BADO]</i>							1/0
1350	Objects in Distributed Systems	1/5						1/1
1348	Specifying Services for Distributed Systems				2/1			1/2
1344	Remote Procedure Call in Distributed Systems						1/4	1/3

Table 1: Course modules by course

APM	Module	TA/ TDAS1 [DSAS for CNET]	[IDP]	IE/ IDPE2	IA/ IDSA2 [IDDS for CNET]	UA/ UDSA3	ADDS for CNET	BD/ BAD03
1331	The Engineering Model						1/5	1/4
1373	DCE Distributed Services						2/1	1/5
1345	The CORBA Object Management Architecture	1/4						1/6
-	<i>Review of Day 1 [BADO]</i>							2/0
1365	Concurrency in Distributed Systems							2/1
1366	Building Applications with ANSA							2/2
M1343	Naming in Distributed Systems							2/3
1341	Dependability in Open Distributed Systems							2/4
1352	Designing Applications with CORBA				2/2			2/5
1349	CORBA Object Services				2/3			2/6
	<i>Review of Day 2 [BADO]</i>							3/0
*1379	CORBA in the Real World							3/1
1358	Replication Techniques for Distributed Systems						1/6	3/2
1357	Exploiting High Performance Networks						2/4	3/3
1353	Real-time Distributed Systems						2/5	3/4
1354	Multi-media in Distributed Systems						2/6	3/5
1356	Course Roundup [for BADO]	-	-	-	-	-		3/6
[1407]	The COM Object Model	1/6						
-	Object-Oriented Methods for Distributed Systems				2/5			
-	Telecommunications Network Management						1/3	
-	Distributed Algorithms						2/2	
-	Persistence in Distributed Systems						2/3	
1313	Enterprise Modelling for Distributed Systems	-	-	-	-	-		-
1363	Development Tools and Methods for Distributed Systems	-	-	-	-	-		-
1359-	Assessing Open Distributed Systems							

a. A short session, only 30 minutes

b. Appears incorrectly as APM.1329 in the proposal (APM.1450)

References

[Background Plan]

Training Project Plan
APM.1260

[Training Plan]

Training Plan
Rob van der Linden

[Course Review]

Initial review of part I and II Training Courses
APM.1432

[Modules]

ANSAwise Training - topics and modules
APM.1273

[Exercises]

ANSAwise Training - case studies and exercises
APM.1401

[ANSA Overview]

An Overview of ANSA
AR.000.00

[Abstract and Automate]

Abstract and Automate
TR.042.00

[DCE Internals]

DCE Internals Course - Student Guide Vols 1 and 2
Open Software Foundation

[ODP Trader]

ODP Trading Function
ISO/IEC DIS 13235

[ANSAware/RT]

Real Time ANSAware
APM.1452

[Trader Presentation]

Trader Presentation to OMG
APM.1447

[CNET Proposal]

Proposal for Training in Distributed Systems
APM.1450

[Impact of Distributed Processing]

**Welcome [to Impact of Distributed Processing]
APM.1471**

[CORBAServices]

**CORBAServices: Common Object Services Specification
OMG Document Number 95-3-31**

[ObjectWorld 95]

Object World UK 1995 Conference Notes

[Total Area Networking]

**Total Area Networking - John Atkins and Mark Norris
John Wiley -BT
ISBN 0-471-95480-2**

[Power Programming with RPC]

**Power Programming with RPC - John Bloomer
O'Reilly & Associates, Inc.**

[Otelso Workshop]

**Workshop on Distributed Object Environments - May 9th, 1995
Otelso (Eureka 1001)**