



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

ANSA Phase III

Initial review of part I and II Training Courses

Yigal Hoffner

Abstract

Part I: "Understanding Distributed Systems Architecture"

Part II: Building Distributed Applications Using Objects

This document outlines the problems and opportunities in the present course material and suggests a plan of action for review and improvements.

APM.1432.00.01

Draft

7th March 1995

Project Management (confidential to ANSA consortium for 2 years)

Distribution:

Supersedes:

Superseded by:

Initial review of part I and II Training Courses



Initial review of part I and II Training Courses

Yigal Hoffner

APM.1432.00.01

7th March 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	Overview of Part I & Part II
1	1.1	Overview of this document
1	1.1.1	The business problem - a warning!
2	1.1.2	The technical problem
2	1.1.3	The solution
2	1.2	Overall work
3	1.3	Review process - per course
3	1.3.1	Introduction
3	1.3.2	Per module
4	1.3.3	Review level
5	2	Presentation/Admin. issues
5	2.1	Presentation
5	2.1.1	Course attendants and second reading
5	2.2	Back up for speaker
5	2.3	Admin.
6	2.4	Site
6	2.5	Questionnaire
7	3	Module comments: "Understanding Distributed Systems Architecture"
7	3.1	Course Timetable - Day 1
7	3.1.1	Welcome
7	3.1.2	Introduction to Distributed Systems
7	3.1.3	Distributed and Networked Operating Systems
8	3.1.4	Characteristics of Open Distributed Systems
8	3.1.5	Templates for Distributed Applications
8	3.1.6	Architecture for Open Distributed Systems
8	3.1.7	Networking in Distributed Systems
8	3.2	Course Timetable - Day 2
8	3.2.1	Review of Day 1
8	3.2.2	Introduction to CORBA and DCE
9	3.2.3	The ODP Reference Model
9	3.2.4	Management of Distributed Networks
9	3.2.5	Distributed Communications Techniques
9	3.2.6	Distributed Database and Distributed Systems
9	3.2.7	The Computational Model
10	3.3	Course Timetable - Day 3
10	3.3.1	Review of Day 2
10	3.3.2	Trading and Federation
10	3.3.3	Service Quality in Distributed Systems
10	3.3.4	Distributed Workflow Applications
10	3.3.5	Security in Distributed Systems

10	3.3.6	Services in the Electronic Marketplace
10	3.3.7	Course Roundup
11	4	Module comments: “Building Distributed Applications using Objects”
11	4.1	Course Timetable - Day 1
11	4.1.1	Objects in Distributed Systems
11	4.1.2	Specifying Services for Distributed Systems
11	4.1.3	Remote Procedure Call in Distributed Systems
11	4.1.4	The Engineering Model
11	4.1.5	DCE Distributed Services
11	4.1.6	The CORBA Object Management Architecture
12	4.2	Course Timetable - Day 2
12	4.2.1	Review of Day 1
12	4.2.2	Concurrency in Distributed Systems
12	4.2.3	Building Applications with ANSA
12	4.2.4	Naming in Distributed Systems
12	4.2.5	Dependability in Open Distributed Systems
12	4.2.6	Designing Applications with CORBA
12	4.2.7	CORBA Object Services
12	4.3	Course Timetable - Day 3
12	4.3.1	Review of Day 2
12	4.3.2	CORBA in the Real World
12	4.3.3	Replication Techniques in Distributed Systems
12	4.3.4	Exploiting High Performance Networks
12	4.3.5	Real-time Distributed Systems
12	4.3.6	Multimedia in Distributed Systems
12	4.3.7	Course Roundup
13	5	Examples
13	5.1	The whole course story
13	5.1.1	For engineers
13	5.1.2	For managers
13	5.2	Another look at the story
14	5.2.1	The Oil Rig story
14	5.2.2	The automation of work environments story
14	5.2.3	Interworking and portability story
14	5.2.4	The electronic market place picture
15	5.2.5	The reversed assumption story
15	5.3	The story for specific modules
15	5.3.1	Objects in Distributed Systems module (Part II)
15	5.3.2	RPC module (Part II)

1 Overview of Part I & Part II

Comments about:

- **“Understanding Distributed Systems Architecture”**.
- **“Building Distributed Applications using Objects”**.

1.1 Overview of this document

- Chapter 1 comments on overall course problems. These should be applied to all present and future courses.
- Chapter 2 provides more detailed comments about presentation/admin. issues.
- Chapter 3 provides specific comments per module. Also see comments in course notes.
- Chapter 5 provides examples relating to both courses as to what is meant by a “story” and the importance of the logical flow.

1.1.1 The business problem - a warning!

The real test of the courses is when you get a few people attending who come from a small company or who pay their own fees as opposed to a situation where employees of a large company attend, particularly one with a 2 billion pound profit.

Lack of resources to do the job properly will ultimately result in dissatisfied customers and speakers.

This is not a criticism of Chris Mayers’ effort - i think he has done an excellent job but that was only the first draft and it requires a proper review.

When i volunteered to help with the training i was under the impression that the necessary resources will be allocated to do the job.

The broad review of each TC presentation includes:

- the overall program is prepared by or with RVDL who tries to put an overall story on the progress and tries to show how the different parts are coming together
- each presentation is rehearsed in front of at least 3 people
- each presentation looked at and is approved by AJH

The TC is a restricted audience. By and large they are research people, they know us and what we are trying to do, and they are our supporters. A wider audience such as, for example, people from development labs may have more influence than the TC members and they are not automatically our supporters. We should be seriously worried about what they are presented with and with our professional image. Ultimately, these courses may affect the research program as well as the business unit through our collaborators.

When the courses are given to companies other than our collaborators, the reputation outside the consortium may suffer.

We are reaching the point where providing “anything” on distributed systems is no longer good enough. Given that there is now some choice - people will want good value for money.

Consequently, each course module should be given **at least** the same treatment as that of a TC talk, but preferably much more.

1.1.2 The technical problem

To review and update the course notes and to modify them so that **more** than one person can give these courses. There are parts of the courses which everyone at APM should be happy with and should be able to give (introduction to Part I and Part II).

1.1.3 The solution

- A top down approach to reviewing the courses
- The approach i am outlining describes a process, similar to the review process of documents at APM, which should be applied rigorously to the courses
- Not all the things stated below are entirely missing at the moment. Reader must remember that the work put into the courses and the results obtained by Chris are amazing. There is, however, always room for improvement, particularly in view of the warning above.

1.2 Overall work

1. courses which fit together as a sequence must be considered both separately and together
2. define audience classes which make a difference as to what the presentation should include, how it should be presented
3. provide a coherent statement of the objectives of each course in context of audience group:
 - for course providers
 - for course attendee - to be stated in the course manual.
4. the statement of the objectives of the course will serve as a starting point for determining the:
 - content
 - structure of the course
 - the way its presented
5. this can be used to break the content into modules
6. provide a story: a coherent description of the flow of the course, how do the units fit together. There has to be a story:
 - for the whole course
 - for each module which fits together. See example section (Chapter 4 and Review of Part II).

7. at the end of the course, participants should be in a position to tell that overall story to their colleagues in more or less detail. This should be one of the objectives of the course
8. presentation of the road map: when put on different pages it is impossible to see the overall structure. It is possible to put pages in the manual which do not fit on a slide. Speaker can refer to these in the talk and listeners can look at manual rather than screen. Put in another way - it should be possible for the speaker to tell the story when going through the course content/structure. Where possible it should provide a classification of issues/approaches/solutions in distributed systems.
9. chart for story + structure in terms of a classification of topics is needed for entire course and for each module.
10. the same treatment should be applied recursively to each module at the appropriate level. Again, a statement of the objectives of each module should be put in start of module. Also how it fits with the rest of the course. An explanation of the structure of the chapter if it is long is also important.
11. special attention an care should be taken with the introductory part.
12. specialization of course for difference audiences (modules can be picked from a central pool - but story must still be worked out for each group):
 - managers
 - engineers
 - other classes of audience
13. After the objective and content are checked for each course it will be necessary to re-check whether the length of the course agrees with content. It is too early to decide but i believe Part I should be cut down to 2 days at most.
14. Clear objectives of course and a statement describing what the attendee will know at the end of it will be useful for people trying to chose which course to attend.

1.3 Review process - per course

1.3.1 Introduction

Statement of objectives and the introduction part should be review by the entire technical team. This should be an introduction that everyone here should be able to give.

Course as a whole to be review at story level and in the context of the objectives.

One person appointed chief reviewer of entire course.

1.3.2 Per module

Each module should be revised separately. I suggest the same treatment given to documents reviewed at APM:

- one person appointed chief reviewer of each module

- presentation to team or at least selected audience of each module - we do this for TC presentations because it is important
- selected review team to comment on slides (at least one expert and one non expert in review team)
- person responsible for module implements recommendations.

1.3.3 Review level

Review should be carried out at different levels:

- **technical** detail: In some places the technical detail is misleading. There has to be a check of that as well. This is ultimately the most important aspect of the presentations.
- **language**: note that a serious review of the use of language has not been conducted. Most of the comments put during course are at the general content level.
- **presentation**: some comments about course length etc. already made. But serious consideration of lay out (see DCE course for example) not done.

2 Presentation/Admin. issues

2.1 Presentation

- Ideal length of sessions is not 60 minutes. Some advice from JM (OU professor) would be useful on optimal length.
- attempt to produce around 30 slides per module no matter what is arbitrary and totally inappropriate. As a result several talks with about 30 slides take anywhere from 27 minutes to 80 minutes.
- time per slide in most cases is too fast
- not all information has to be in slide form. It is possible to have textual hand-out and occasionally even talk around something which is in front of listeners but not on the screen, e.g. road map
- more exercises - particularly design ones
- many slides have too little on them causing two problems:
 - speaker spends hardly any time on them raising the question of whether they are necessary or could be condensed
 - impossible to work out what was said at a second reading

2.1.1 Course attendants and second reading

When looking at the course notes a few weeks later it is impossible to guess what was the message in those slides or even what the overall argument is. It would be good to be able to follow the argument by looking at the slides.

An important question is also how much should someone who has not attended the course be able to get from the notes.

2.2 Back up for speaker

A number of key issues which are covered briefly - if speaker had extra slides they could pull out if necessary. Could be one or more slides per issue or more if necessary. Use team member expertise on the subjects.

Topics (examples)

- technology: MACH, DCE, CORBA, Netware
- concepts: binding, trading, factories, interceptors
- issues: dependability, type system, Real-time, QoS

2.3 Admin.

- name tags for attends would be very useful
- why give the ANSA logo paper in folder - we are not a hotel

- double sided rather than single sided photocopies:
 - the folder is so full that the pages are now tearing apart. It's too heavy anyway
 - hardly anyone takes notes
 - those who do rarely use back side of previous page
 - why not give them A4 paper with holes so they can add it to folder when needed. Seems a simpler solution and certainly more cost effective for everyone
 - course notes can be condensed so that they do not server as a door stopper
 - personally - if i can carry the notes with me i am more likely to look at them at home or at breaks
 - **Think about the trees!**

2.4 Site

Off site boarding places are better from the social point of view and the ability to have evening exercises. From experience the difference between the two is quite astounding.

2.5 Questionnaire

Does it provide adequate cover of the course and provide adequate feedback?
Overall yes, but more specific information is required, per module for example.

3 Module comments: "Understanding Distributed Systems Architecture"

3.1 Course Timetable - Day 1

3.1.1 Welcome

See comments in chapter 1 about:

- the lack of statement of objectives
- what will the listeners know/be able to do at the end of the course
- lack of explanation of an overall structure or a useful road map

Road map does not provide any insight to why the topics were chosen nor to their relative importance and relationships between them. No indication of structure. If this module is just a word about administrative issues, then this should move to next module. Either way all of the above is missing.

3.1.2 Introduction to Distributed Systems

I could not detect a structure in the slides. See comments in chapter 1.

Architecture for reuse is inappropriate. The architecture is for dealing with complexity. Reuse may be a derivative of it. Explanation of architecture is weak.

This part was very weak and also raised comments from attendee.

3.1.3 Distributed and Networked Operating Systems

This is important. This is a history story and should be introduced as such not as networked operating systems. This is another example of lack of clear objective. I would rename the module to: "History/Evolution of computer systems (with a view towards distributed systems)". The story is one of a continuous transition, the mess we are in and that we are not at the end of the road yet (if there is one). The themes are both technical and business.

Technical story: more MIPS, more comms, more bandwidth, better tools, larger scale and scope etc.

Business/enterprise story: transition from one person - one machine as an extension of the pocket calculator, type-writer and file cabinet, to more and more of the business aspects and the communication and co-operation between people being subsumed in the system.

The material on micro kernels and multicomputers is unnecessary: at most one slide mentioning some trends can be made but this is immaterial.

3.1.4 Characteristics of Open Distributed Systems

Example and exercise of "openness" very good! The rest do not fit with openness at all.

List of characteristics incomplete - why?

Principles mentioned are not exactly principles and fall short of explaining anything at this point, and leave me hanging thinking - so what?

Complete revision of this module and try and see where openness fits into.

3.1.5 Templates for Distributed Applications

This gets worse in the next chapter where i cannot see what templates are in this context, nor what the objective of the module is. I cannot tell whetehr3 this is still part of the problem statement or solution. It was indicated in the last module that we are approaching a solution in the form of the work at ANSA.

3.1.6 Architecture for Open Distributed Systems

Incoherent structure. Better than previous modules but hard to say as overall structure has no apparent logic. Abstract and Automate should be the basis for this.

3.1.7 Networking in Distributed Systems

Why this module after introducing architecture? Completely out of place. No flow of story.

Not sure what this module is trying to achieve. Is it trying to show that doing comms yourself is complicated? That we need platforms/transparency?

Link to distributed system is missing completely. There is no context to the talk.

The whole idea of distributed application platforms is to avoid as much as possible the complexity of comms. This is not explained anywhere in the module.

3.2 Course Timetable - Day 2

3.2.1 Review of Day 1

Too quick and cursory. Again if there was an overall story it would be easier to do this.

3.2.2 Introduction to CORBA and DCE

Relationship between architecture and platforms not explained.

A diagram could be useful here showing relationship between ANSA - ANSAware, OSF- CDE and OMG - CORBA. Can summarise about half the slides in this module if this is what you are trying to show. I cannot understand why this would interest people at this level of detail.

This has more on OMG than on CORBA. What is actually said about CORBA does not convey much information.

3.2.3 The ODP Reference Model

Why ODP here - as an example of architecture, or an attempt to standardize on one? Why is it between Introduction to CORBA and DCE and Management of Distributed Networks.

Too many slides (45) - far too long for what the objectives state. decide what is the point here - an example architecture, an example of a standardization process?

How will ODP be used - the benefits, the method, etc.all missing.

Is this module needed at all?

3.2.4 Management of Distributed Networks

Why jump back to networks and their management. Looks like a criss cross of a bit of architecture, followed by Networking in Distributed Systems, then a bit of platforms, followed by Management of Distributed Networks, then Distributed Communications Techniques. Distributed systems are completely lost in the first half of the second day.

This topic is important and RPC use in the solutions should be emphasized here. Again, if this is not put in the context of distributed systems than why are we giving this course under the above title?

3.2.5 Distributed Communications Techniques

I fail to see the message or the need for this module.

I don't understand the link between the Prussian general problem and the rest of the chapter.

RDA is important but not here. Possibly next chapter is better context for it.

This module should be dropped or seriously revised.

3.2.6 Distributed Database and Distributed Systems

This chapter is not about Distributed Database. It is about RDA. What is the aim of the module then?

3.2.7 The Computational Model

ANSA'ish from now on - Indicate the course is shifting here to another angle.

This and the next module are reasonable but again, it is hard to judge as there is no overall structure and no way of telling why this is done at this point. It has very little relation to what is before it and only some to the next module.

What is missing is a link between Computational view and engineering, tying this to RPC at some level. This is because engineering is left for the next course. maybe it should be presented here after all.

It will be more appropriate to have the module from Part II on Object orientation.

3.3 Course Timetable - Day 3

3.3.1 Review of Day 2

Too quick and cursory.

3.3.2 Trading and Federation

This part worked very well when the examples were discussed - see slide 11. It is worth checking whether it is not possible to get this kind of discussion going more often.

This says very little about federation - it is not appropriate here. Scaling is not the reason why we have federated systems.

3.3.3 Service Quality in Distributed Systems

Initial exercise very good.

Some detail is very confusing in later slides. Order of slides is not clear to me in places. See comments on slides.

3.3.4 Distributed Workflow Applications

Not sure at all why this is in the course at all. This module should be dropped.

3.3.5 Security in Distributed Systems

I like this module. It seems to lack structure but it's nevertheless good. Again - the introduction should provide some structure for the listener and reader.

Slides 40-end should be in an appendix.

3.3.6 Services in the Electronic Marketplace

This seems to me to fit better in day one as part of an introduction as to where things are going. It misses the point in day 3 and there is no good tie in to what was discussed in the 2 and half days before.

So either leave and tie in with previous material, or transfer to start of course (day 1).

I like the examples concerning remuneration.

Too many slides.

3.3.7 Course Roundup

Not convinced about this part but that is likely to be a reflection of the problem outlined with the introduction to Part I.

4 Module comments: "Building Distributed Applications using Objects"

4.1 Course Timetable - Day 1

I could not detect an overall structure in course - again there is no story. See comments in chapter 1 about:

- the lack of statement of objectives
- what will the listeners know/be able to do at the end of the course
- lack of explanation of an overall structure or a useful road map

The "slide About this course" first bullet is incorrect. The course does not provide practical techniques and advice for building distributed applications, with the tools and products you can buy today.." - no it does not. At best it gives some overall view of a few platforms currently available. This is far from "... advice or practical techniques for building distributed applications".

4.1.1 Objects in Distributed Systems

This is not a bad module but it requires a lot of work. Again, no overall structure and it shows.

Lack of material on object pointers (interface references) causes problems in later modules when it comes up unexpectedly in next module.

This should be tied together with some material from the Computational Model module in Part I.

4.1.2 Specifying Services for Distributed Systems

4.1.3 Remote Procedure Call in Distributed Systems

4.1.4 The Engineering Model

4.1.5 DCE Distributed Services

4.1.6 The CORBA Object Management Architecture

4.2 Course Timetable - Day 2

- 4.2.1 Review of Day 1**
- 4.2.2 Concurrency in Distributed Systems**
- 4.2.3 Building Applications with ANSA**
- 4.2.4 Naming in Distributed Systems**
- 4.2.5 Dependability in Open Distributed Systems**
- 4.2.6 Designing Applications with CORBA**
- 4.2.7 CORBA Object Services**

4.3 Course Timetable - Day 3

- 4.3.1 Review of Day 2**
- 4.3.2 CORBA in the Real World**
- 4.3.3 Replication Techniques in Distributed Systems**
- 4.3.4 Exploiting High Performance Networks**
- 4.3.5 Real-time Distributed Systems**
- 4.3.6 Multimedia in Distributed Systems**
- 4.3.7 Course Roundup**

5 Examples

Note: I can say more on this at a white board!

It is not suggested that all the pictures have to be used everywhere - just that they provide the basis for providing overviews of the subject and can be used both at course level and at module level.

A good story can be used to generate the topics and be used as a check list for completeness.

5.1 The whole course story

An overall story for any type of audience is that of the reality of distributed systems etc.

The engineers story and managers story are not exclusive. that is, you should outline to managers what are the difficulties facing engineers and vice versa. But the emphasis will be different for each group.

Please note that this is only an example!

5.1.1 For engineers

The reversed assumption scenario can provide the overall story i.e. the paradigm shift from centralized to distributed systems.

No need to convince this audience that distributed systems are here, are important and are exciting. Message to get across is that of the solution, not the problem. The solution is outlined in A&A paper.

The technical problems: learning to think differently. Using new concepts and tools. Not a piecemeal approach but a coherent one, i.e. an architecture and the appropriate tools!

5.1.2 For managers

What are the problems when you connect systems together - what are the things that can be different.

The business problems: Jane's story. Need to convince this audience of the scale of the problem, it's ubiquitousness: ducking won't help.

The problem and solution is outlined in A&A paper.

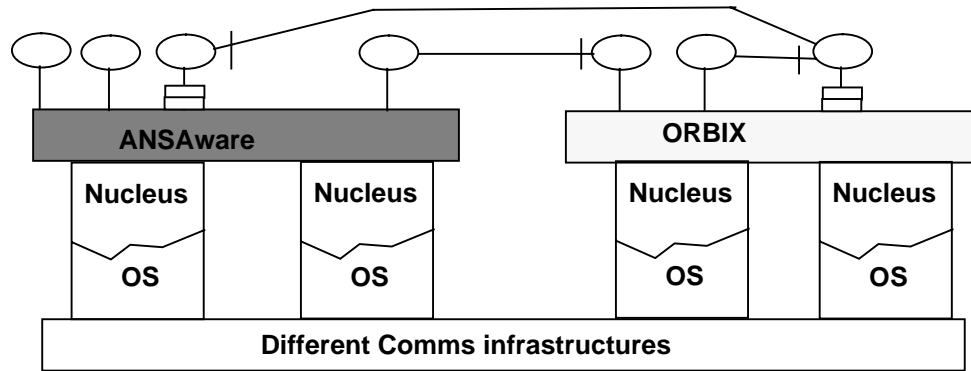
5.2 Another look at the story

Do a brain chart for whole course at appropriate level and for each chapter - like rvdI always does.

5.2.1 The Oil Rig story

This allows showing the platforms and talking about the relationships between platforms and OS's and comms. Fits with a historical view of the field but also when explaining the relations between different platforms.

Figure 5.1: The old Oil Rig picture



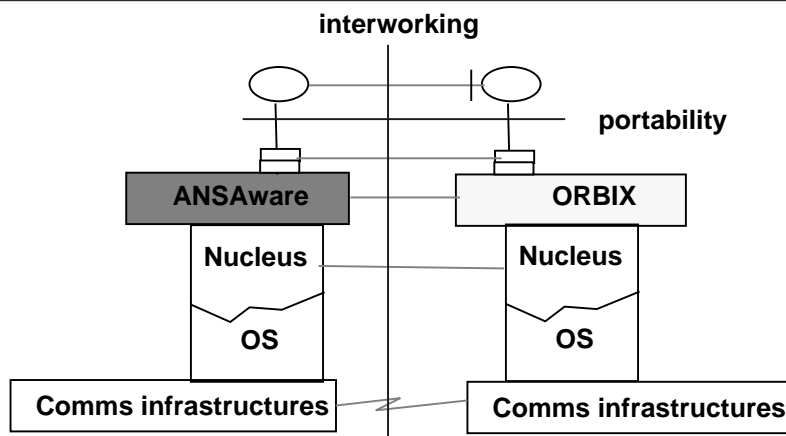
5.2.2 The automation of work environments story

1. From one person - one PC where communication is done by people outside the system
2. To networks allowing some limited channels of communication
3. To distributed systems allowing:
 - Personal: group work: multi-conferencing, group editing, diaries, etc.
 - Organizational: the electronic market place
4. future: ?

5.2.3 Interworking and portability story

From the construction part of the computational model.

Figure 5.2: Interworking and portability story



5.2.4 The electronic market place picture

See Nigel's picture which is also in electronic market place module in Part I.

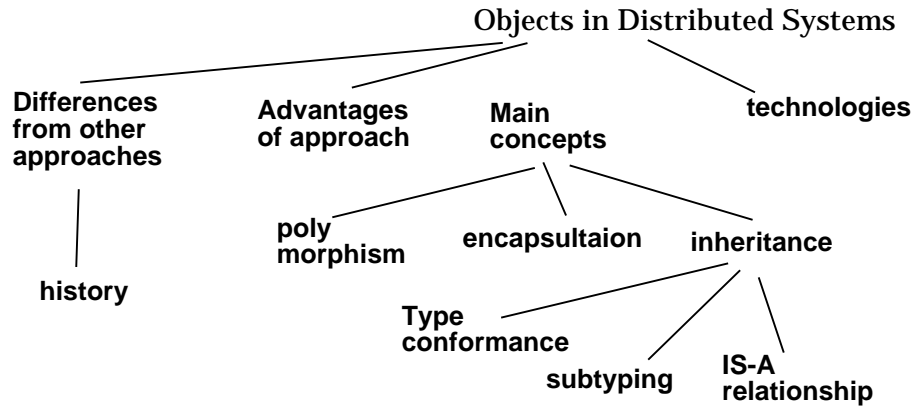
5.2.5 The reversed assumption story

See JPW edited paper “Introduction to the ANSA architecture” [?].

5.3 The story for specific modules

5.3.1 Objects in Distributed Systems module (Part II)

Figure 5.3: Suggestion for module content and structure



5.3.2 RPC module (Part II)

Three pictures can serve as a starting point:

- client, server interface configuration (Figure 5.4)
- IDL to stub generation tool chain (Figure 5.4)
- example of an interface (Figure 5.4)

Figure 5.4: client, server interface configuration

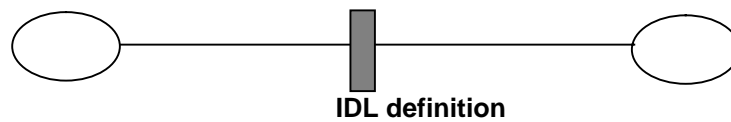


Figure 5.5: IDL to stub generation tool chain story

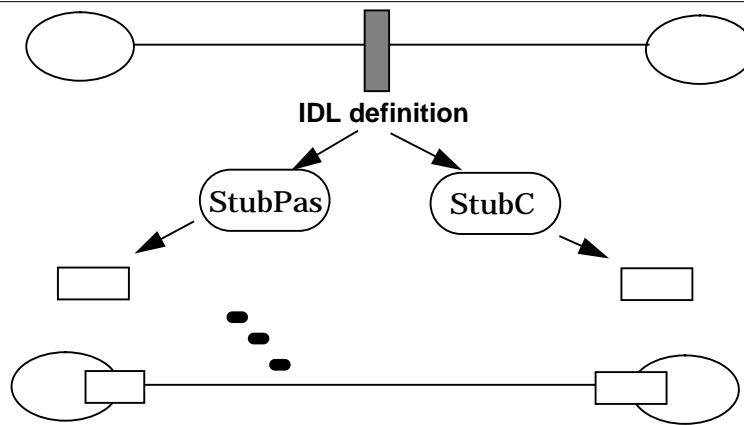


Figure 5.6: IDL and type conformance

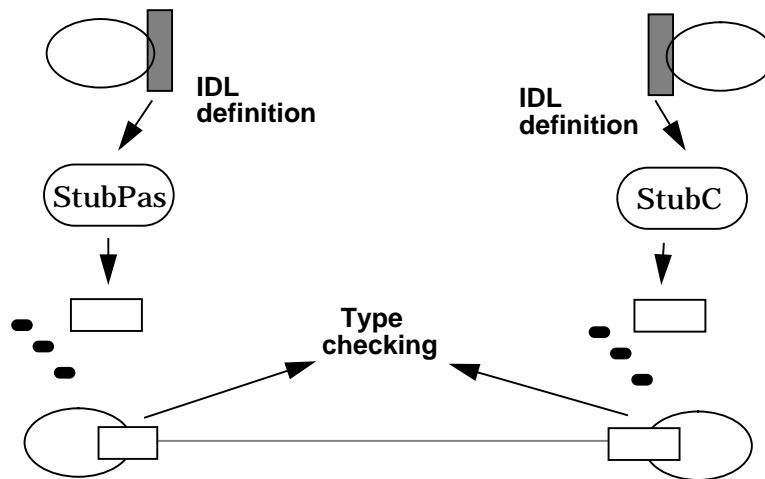
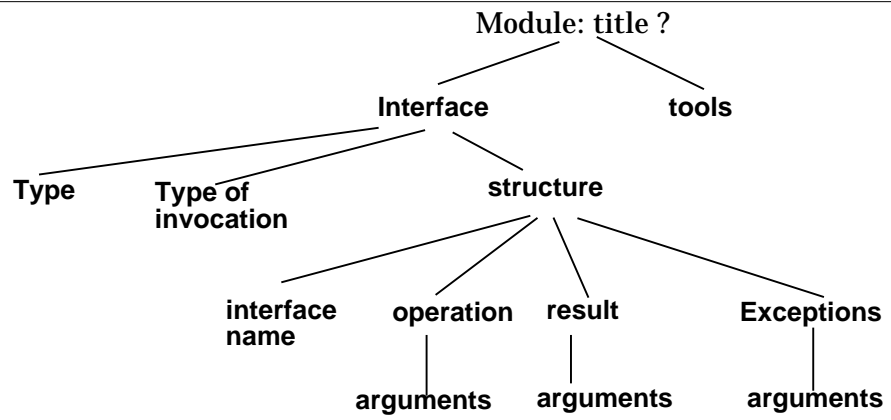


Figure 5.7: example of an interface

```

Interface X:
  OP1(arg1, arg2)
  Return (arg1, arg2)
  exception
    E1(arg1, arg2);
    E2(arg1, arg2);
    
```


Figure 5.8: Suggestion for module content and structure



References

[LINDEN 93]

van der Linden R. J., *An Overview of ANSA*; **AR.000.00**, APM Ltd., Cambridge U.K., May 1993.

