Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom

TELEPHONE: **Cambridge (0223) 323010**
INTERNATIONAL: **+44 223 323010**
FAX: **+44 223 359779**
E-MAIL: **apm@ansa.co.uk**

ANSA Phase III

Open Dependable Distributed Systems

Nigel Edwards

Abstract

The basic technology for open distributed processing is now understood. The challenge now is to be able to deliver appropriate non-functional guarantees (e.g. reliability, availability and performance), and to be able to integrate existing services and systems into this world of open dependable distributed computing. It is important to realize that there will not be one set of non-functional guarantees which are appropriate to all applications; any solution must allow the selection of guarantees to match different application requirements. Until this challenge is met, open distributed computing will not be used in business critical applications.

This document introduces and sets the scope for the ANSA work on dependable distributed computing. It looks at the need for Open Dependable Distributed Systems and explains the ANSA vision for the development of such systems. Central to this vision is the concept of selective transparency. ANSA is based on a number of principles, the consequences of these principles for dependability are examined.

APM.1145.01 **Approved** 14 March 1994
Briefing Note

Distribution:
Supersedes: APM.1073
Superseded by:

Open Dependable Distributed Systems

Open Dependable Distributed Systems

Nigel Edwards

APM.1145.01

14 March 1994

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	(0223) 323010
INTERNATIONAL	+44 223 323010
FAX	+44 223 359779
E-MAIL	apm@ansa.co.uk

The sponsors of the ANSA Workprogramme have agreed to allow access by companies which have signed an agreement with Bellcore in respect of the Workprogramme of telecommunications research currently known as TINA-C to permit said companies access to and use of certain documents, software, information and deliverables arising from the results of the ANSA Workprogramme. This information will made available by the ANSA sponsors either as paper copies or through the medium of electronic file transfer from the information storage system operated by Architecture Projects Management Limited on behalf of the ANSA sponsors.

This is one such document and access is allowed in strict confidence on the understanding that the user accepts these conditions and on the sole basis that it will be restricted to those persons involved in the DPE work package of the TINA-C Workprogramme and that it will not be disclosed to any other person, firm or corporation.

The use of this information is restricted to its use only for the purposes of the carrying out of the DPE workpackage of the TINA-C Workprogramme and only at the site provided by Bellcore for that Workprogramme. No licence or permission for its use in any other part of the TINA-C Workprogramme or for its subsequent exploitation is granted and the ownership and copyright of all such documents, software, information and deliverables is expressly retained by Architecture Projects Management Limited for and on behalf of the sponsors for the time being of the ANSA Workprogramme. In the event of a company leaving the TINA-C Workprogramme or resigning from its bilateral agreement with Bellcore, then that company shall promptly and without demand return to Architecture Projects Management Limited all copies of any information, documents, software or other IPRs obtained under these provisions.

The access granted by these provision is on the understanding that the TINA-C consortium and the sponsors for the time being of the ANSA Workprogramme intend to and shall promptly enter into a suitable formal agreement for access to information and interavailability of IPRs (including software) for the purposes of the carrying out of the ANSA and TINA- C Workprogrammes.

With regard to any company which is participating in the TINA-C Workprogramme and which is also a sponsor of the ANSA Workprogramme, the obligation of confidentiality and the use restrictions contained in these provisions shall be subject and without prejudice to the obligations undertaken by, and the rights granted to, such company under the ANSA sponsorship agreement.

1 Open Dependable Distributed Systems

1.1 Introduction

Dependability is increasing in importance in the market place. A recent Gartner report predicts the market for fault-tolerant systems will double in the next three years (from ‘mid 1993) [GARTNER]. In an increasingly fierce market, reliability and availability can have significant effects in reducing the cost of ownership [SIEWIOREK 92], thus giving a vendor a competitive advantage. Within the context of large distributed systems, dependability will be particularly important: the more components a system has the greater the probability that one of those components will be faulty. In addition, openness further reduces the cost of ownership by allowing easy integration and incremental evolution of the information system [HERBERT 93], [HARRIS 93].

The basic technology for open distributed processing is now understood: there are now several de-jure and de-facto standards which are emerging. The challenge now is to be able to deliver appropriate non-functional guarantees (e.g. reliability, availability and performance), and to be able to integrate existing services and systems into this world of open dependable distributed computing. It is important to realize that there will not be one set of non-functional guarantees which are appropriate to all applications; any solution must allow the selection of guarantees to match different application requirements. Until this challenge is met, open distributed computing will not be used in business critical applications.

This document introduces and sets the scope for the ANSA work on dependable distributed computing. Section 1.3 looks at the need for Open Dependable Distributed Systems (ODDS). The emphasis is on the need for dependability, readers requiring justification for Open Distributed Systems are referred to [HERBERT 93]. Section 1.4 explains the ANSA vision for the development of ODDS. Central to this vision is the concept of selective transparency: different applications will have different dependability requirements, requiring the selection and configuration of different dependability mechanisms. ANSA is based on a number of principles described in [LINDEN 93]; §1.5 examines these principles and what they reveal within the context of dependability.

1.2 Context and audience

The audience for this paper are managers, engineers and system designers who need to understand the benefits that will result from the ANSA work on dependability. This paper has a companion paper intended for system designers and engineers which looks at the basic concepts used to describe and build ODDS [EDWARDS 94].

1.3 The need for open dependable distributed systems (ODDS)

This section looks at the need for dependability in open distributed computing, the advantages to be gained by delivering the right solution quickly, and some of the constraints on the technology used to deliver ODDS.

1.3.1 Business critical applications need dependability

Deploying a business critical application or information service without any guarantees about the dependability of that application is analogous to participating in a business transaction without any formal contractual arrangements. It may be fine, but the consequences of failure could be severe. In the absence of any contract to set expectations, there is more chance of something unexpected happening — something which may be viewed by one of the parties involved as a failure.

Similarly it could be disastrous for a business to rely on an application without well defined expectations — without clearly defined dependability guarantees. Hence exploiting open distributed computing to deliver business-critical information services will require being able to offer both functional and non-functional (dependability) guarantees which are appropriate to the information service being provided.

1.3.2 Current technology does not address dependability

There are a number of de-facto and de-jure standards emerging which incorporate technology for distributed computing: ODP [ODP 93], CORBA [OMG 91], Atlas [UI], DCE [OSF 91] (including DME and ENCINA [SHERMAN 93]) and the various OSI standards (e.g. GDMO [GDMO], OSI RPC [OSI RPC], OSI TP [OSI TP] etc.). All of these provide applications (objects) with a means of communication. Perhaps the highest level of functionality is delivered by CORBA which supports object based distributed computing: objects can invoke each other regardless of whether or not they are co-located. In addition all these standards identify some basic services which are needed by applications, such as naming. Hence the technology for delivering basic “open distributed computing” is becoming well understood and standardised.

With the exception of ODP (which has been heavily influenced by previous ANSA work), very little work has been done on providing appropriate dependability guarantees [HERBERT 93]. ODP with its notions of transaction, group and replication transparencies lays some of the foundations [ODP 93].

1.3.3 Gain a competitive advantage: match customer requirements

One of the basic principles of ANSA is that different customers and different applications will have different dependability requirements. Even within one application the different components will have different availability, reliability and consistency requirements [CAMERON 93]. Understanding the engineering and cost trade-offs in building dependable distributed systems will enable the vendors to match the dependability delivered to the requirements of the customer and the application, giving them a competitive advantage over those who cannot do this.

1.3.4 Gain a competitive advantage: deliver the solution quickly

Competitive advantages are also gained by being able to deliver the right solution more quickly than the competition. One way of doing this is to

minimise the amount of bespoke engineering in a solution. The approach should be to use tools, configuring basic standard engineering components to deliver the guarantees which are needed by the application.

1.3.5 The need to incorporate existing systems into ODDS

The need to preserve investments in existing information technology infrastructure, means that new information services will have to interwork with so-called legacy systems and yet still provide a some guarantees about dependability. This means that there will be few opportunities to build systems from scratch; rather, it will be important to understand how to configure mechanisms to get appropriate non-functional guarantees from what already exists. Openness implies the ability to be able to cope with heterogeneity at all levels: different machines using different operating systems interworking between different administrations.

1.3.6 Hardware versus software techniques

The ANSA work on dependability is about developing concepts which can be used for open dependable distributed computing. It aims to put in place the technology which enables the construction of information services with various dependability guarantees [CAMERON 93]. Since openness implies minimising the assumptions about the underlying hardware and operating system, this work concentrates on software rather than hardware techniques for dependability, and on techniques which do not require one particular underlying platform for distribution.

1.4 Developing ODDS using selective transparency

This section describes the ANSA vision for developing ODDS and discusses how this relates to what is available now.

1.4.1 The vision

Consider how a dependable information service might be developed in an ideal world. First a a precise statement of the correct behaviour of the service is needed. This would include a detailed statement of the functional and non-functional requirements of the service (including the dependability requirements). The next step would be to identify the major application-level components of the service including any existing or legacy systems it must interwork with. A precise statement about the dependability of each component and the expected (correct) behaviour is made. The ANSA failure model allows the dependability of these components to be matched to the dependability of the whole service.

Dependability is an application-level concept: the infrastructure cannot be configured to deliver the required dependability without understanding the application semantics [CHERITON 93]. The aim of the ANSA work on programming and transaction models for dependability is to understand what concepts can be used by the programmer to express the dependability requirements implied by the application semantics. In addition there will be trade-offs to be made over what dependability guarantees are provided by the application components and what guarantees are provided by the supporting engineering mechanisms [EDWARDS 94]. The ANSA programming, engineering

and transaction models will help the system designer to understand and make these trade-offs.

Next the engineering mechanisms needed to support the application-level components are put in place, enhancing the functionality provided by the underlying platform (such as DCE or CORBA). This is where the concept of selective transparency is used: selecting and configuring the engineering mechanisms to match the requirements of the application-level components. This is done automatically by tools using the description of the requirements of the application-level components expressed using the programming and transaction models. The tools reflect the ANSA engineering model for dependability which guides the choice of the engineering components, identifying the trade-offs made in choosing one component over another. The ANSA failure model allows the dependability of the engineering components to be matched to the dependability of the application-level components which they support.

The engineering mechanisms include components which continuously monitor and manage the dependability being provided by the application-level components. These mechanism are responsible for diagnosing faults and taking corrective action to maintain the required dependability. The choice and configuration of these mechanisms is guided by the ANSA management model for dependability.

The aim of the ANSA work on dependability is to put the technology in place for achieving this vision. Ultimately it may be possible to describe the dependability of a system in terms of the dependability and composition of its most basic components; this will require iteration across various levels of abstraction and the various viewpoints. The complexity of open distributed systems means that a completely formal or rigorous method is unlikely to be achievable in the near future. Rather the relationships will be enshrined in the rule, recipes and guidelines which constitute the architecture.

1.4.2 The state of the art now

This vision extends far beyond the current state of the art in which programmers are provided with a tool-kit of protocols [BIRMAN 87], or are able to change the behaviour of a particular mechanism by providing it with a different policy [OSKIEWICZ 93]. It is about enabling system designers to make the trade-off between what dependability guarantees need to be provided by the application and what guarantees are provided by the supporting engineering. It is about letting application programmers use a set of simple concepts to declare their dependability requirements and then using tools to match these on to a rich set of mechanisms quickly and efficiently, exploiting various redundancy and consistency techniques (see [SMETHURST 93] and [SIEWIOREK 92] for lists). This vision recognises that there are fewer and fewer opportunities to engineer systems “from scratch”, rather programming will become more and more about adding and configuring new services interworking with existing ones and being able to get the right behaviour (both functional and non-functional) from existing services.

1.5 The ANSA principles and dependability

[LINDEN 93] describes the principles of ANSA. This section looks at those principles which are particularly relevant to dependability; the principles are divided into seven categories — each category is considered in turn.

1.5.1 Separation

Systems should be designed so that separation amongst their parts can be achieved; this means that they can be more flexibly configured. However, this can have the effect of introducing more components reducing the dependability of the system.

Separation means that services may be remote. This introduces the possibility of partial failure: a failure may occur in a remote service request even though the requester's local system has not failed.

The ANSA work on dependability aims to ensure that the required dependability can be achieved in spite of the effects of separation.

1.5.2 Diversity

Large distributed systems will include many significantly different individual systems. This means that data will be widely distributed with multiple representations and different consistency requirements. It is inevitable that different standards and different dependability mechanisms will be adopted in different parts of the system; designers need to be ready for this and the dependability mechanisms need to allow it.

1.5.3 Scaling

The dependability mechanisms used in a system must not impose constraints on the extent to which it can be interconnected and its applications made to interwork. Scaling is about scaling up and down: mechanisms which are efficient in large systems should be designed so they are efficient in small systems or else should be replaceable by mechanisms which are efficient in small systems.

In large distributed systems it is very difficult to implement a notion of universal time or an observer which can observe every event. This means that technologies which assume a global clock or a global ordering on events are not appropriate.

Larger systems will contain more components which increases the probability that there are one or more faulty components in the system.

1.5.4 Federation

Federation deals with heterogeneous authority and how to retain local control in a large distributed system spanning boundaries of authority. The consequences of federation mean that objects are responsible for their own dependability. In addition objects will need to negotiate contracts with objects subject to other authorities: there may be no common higher authority which lays down what the contract should be. The contract will state what each object is entitled to expect of the other (i.e. what the "correct behaviour" should be).

1.5.5 Transparency

A property of a system is transparent if application programmers need not be concerned with it. The aim of the ANSA work on dependability is to hide the details of the dependability mechanisms from the application programmer.

Previous experience suggests that it is possible to make dependability mechanisms such as replication completely transparent to the programmer [OSKIEWICZ 93]. However, there are limitations on making dependability fully transparent. These are explored in [EDWARDS 94].

There is no universal set of requirements for dependability hence, there is no universal configuration of mechanisms. This means that transparency must be selective: programmers can select and configure the mechanisms which are most appropriate to the job at hand.

Selecting and configuring the appropriate mechanisms is likely to be a complex and error prone task. Programmers may well be tempted to implement their own mechanisms, ignoring the ones provided, because they are too difficult to understand and use. This means that programmers must at least be given guidance on how to select and configure mechanisms to match the requirements of their programs. Where possible, tools should be provided to configure and select the mechanisms (this is automated transparency).

Ideally the dependability requirements should be declared as attributes of the object; tools would then configure the most appropriate mechanisms. The difficulty of capturing requirements and the lack of tools which work directly from them, means that programmers will probably have to specify specific mechanisms. Automated transparency techniques will configure the specific mechanisms selected. The programmer is protected from the details of the mechanisms (e.g. see [WARNE 93]).

1.5.6 Concurrency

Concurrency is inevitable in distributed systems. This means that there is potential for conflicting inconsistent changes to be made to data. Mechanisms are needed to prevent this.

1.5.7 Configuration

Systems evolve over time: new parts are added and old parts are removed. ANSA advocates detection and correction of faults as early as possible, ideally before a new component is configured into the system. This limits the potential for a fault in one component to cause damage to the rest of the system. To achieve this in a dynamic system, the description (of the correct behaviour) of a component must be on-line. Such descriptions will form the basis of the contracts described in §1.5.4 and are important in fault diagnosis.

1.6 Summary

There are several de-jure and de-facto standards which are emerging which will provide the technology to make open distributed computing possible. However, for open distributed computing to be exploited in business-critical applications more technology is required. In particular there is a lack of technology which will enable services to be delivered with appropriate and defined dependability guarantees. The ANSA work on dependability aims to address this problem.

It is important to realise that there is not going to be one set of dependability requirements. Rather the dependability requirements will be determined by the application semantics. The concept of selective transparency can be used to select an appropriate set of engineering mechanisms and configure those mechanisms to satisfy a particular requirement. This needs to be automated.

The ANSA principles reveal a number of issues for dependability in open distributed systems.

- Objects are responsible for their own dependability; contracts must be used in a federated environment to state what each object is entitled to expect of others.
- Technologies based on a notion of a global observer or global clocks are not appropriate: they cannot be realised in large distributed systems.
- Faults should be detected as early as possible, ideally before a component is installed into the system.

1.7 Acknowledgement

The author is grateful to Owen Rees of APM Ltd., for his comments on an earlier version of this document and for discussions with him on its content. Comments by Paul Vickers of Hewlett-Packard were also very useful.

References

[BIRMAN 87]

Birman, K.P., Joseph, T.A., “Reliable Communication in the Presence of Failures”, ACM TOCS, Vol. 5, No. 1, p47-76, 1987.

[CAMERON 93]

Cameron, E.J. , “Scenario”, APM.1064, APM Ltd., Cambridge U.K., October 1993.

[CHERITON 93]

Cheriton, D., Skeen, D., “Understanding the Lmitations of Causally and Totally Ordered Communication”, in Proc 14th ACM Symposium on Operating System Principles, 1993.

[EDWARDS 94]

Edwards, N.J., “Building Dependable Distributed Systems”, APM.1144, APM Ltd., Cambridge U.K., 1994.

[GARTNER]

Figures quoted and attributed to the Gartner group in “Supply and demand”, J. Kaye, Informatics September 1993.

[GDMO]

“Guidelines for the Definition of Managed Objects”, ISO/IEC 10165 Part 4.

[HARRIS 93]

Harris, R.J., Fraser, R.J.C., “Command and Control Infrastructures: The need for Open System Solutions”, Keynote Address, IEE International Workshop on Systems Engineering for Real Time Applications, 13 -14 September 1993.

[HERBERT 93]

Herbert, A.J., “Open Distributed Processing — the Solution to a Business Need”, APM.1055, APM Ltd., Cambridge U.K., 1993.

[LINDEN 93]

van der Linden, R., “An Overview of ANSA”, AR.000.00, APM Ltd., Cambridge U.K., May 1993.

[ODP 93]

“Basic Reference Model of Open Distributed Processing”, ISO/ IEC JTC1/SC21, American National Standards Institute, New York, USA, 1993.

[OMG 91]

The Common Object Request Broker: Architecture and Specification, OMG Document Number 91.8.1, August 1991.

[OSI TP]

“OSI Distributed Transaction Processing (OSI TP), IOS/IEC 10026.

[OSI RPC]

"Open Systems Interconnection — Remote Procedure Call", ISO/IEC 11578 (draft).

[OSF 91]

Introduction to OSF DCE, Open Software Foundation, December 1991.

[OSKIEWICZ 93]

Oskiewicz, E., Edwards, N.J., "A Model for Interface Groups", AR.002.01, February 1993, APM Ltd., Cambridge U.K.

[SIEWIOREK 92]

Siewiorek, D.P., Swarz, R.S., "Reliable Computer Systems — design and evaluation", Digital Press, 1992.

[SHERMAN 93]

Sherman, M., "Distributed Transaction Processing in a DCE Environment with Encina", Tutorial presented at 13th International Conference on Distributed Computing Systems, Pittsburgh, USA, May 1993.

[SMETHURST 93]

Semthurst, R., Wharton, P., "OPENframework Availability", Prentice-Hall, 1993.

[WARNE 93]

Warne, J.P, Rees, R.T.O, "ANSA Atomic Activity Model and Infrastructure", AR.004.01, January 1993, APM, Ltd., Cambridge, U.K.

[UI]

"UI ATLAS Distributed Computing Architecture: A Technical Overview", Unix International.